

THIẾT KẾ WEB

BỘ MÔN KHOA HỌC MÁY TÍNH



NỘI DUNG

1. Tổng quan về internet
2. HTML5
3. CSS
4. JavaScript
5. Bootstrap

[Xem đề cương chi tiết](#)

2

TÀI LIỆU THAM KHẢO

- [1] Web Programming with HTML5, CSS, and JavaScript
by John Dean (z-lib.org)
- [2] Khoa Công nghệ Thông tin, Bài giảng Thiết kế Web,
Đại học Công nghiệp Thực phẩm TP.HCM, 2021.
- ASP.NET <http://www.asp.net>
- HTML, CSS, JavaScript <http://www.w3schools.com>
- CSS Layout <http://www.learnlayout.com>

3

KIỂM TRA – ĐÁNH GIÁ

- Chuyên cần : 10%
- Quá trình: 60% (Trung bình 03 bài kiểm tra)
- Đồ án: 30% → yêu cầu đồ án đính kèm

4

PHẦN MỀM SỬ DỤNG

- Môi trường thiết kế :
Adobe Dream weaver CS
Notepad++
- Môi trường lập trình:
Visual Studio .NET 2010/2012
- Web Browser
IE, FireFox, Chrome...
- Phần mềm hỗ trợ
 - Chỉnh sửa hình ảnh: Photo Impact, Photoshop, Paint shop pro, Illustrator, ImageReady, UleadCool 3D, Firework
 - Tạo logo, banner, menu: Xara 3D, Xara Webstyle, Button Animation, DHTML menu, Corel Draw, Button Studio,
 - Tạo ảnh động: Swish MAX, Flash MX, GIF movie, Banner GIF Animation v.v...



5

Chương 1.

TỔNG QUAN VỀ INTERNET

- Giới thiệu lịch sử
- Các khái niệm căn bản
- Các dịch vụ cơ bản của Internet
- Khai thác tài nguyên trên Internet

6

1. Giới thiệu lịch sử

- Mạng Internet xuất phát từ mạng ARPANET của Mỹ (1969) – mạng giữa các trường ĐH.
- 1989 - Tim Berners-Lee phát minh ra giao thức World Wide Web, sau đó tạo ra ngôn ngữ Hypertext Markup Language - HTML
- 1990 – Tim viết trình duyệt (Web Browser) và Web server đầu tiên (info.cern.ch)
- Từ năm 1993 Internet phát triển rất nhanh
- 1.12.1997 Internet chính thức được cung cấp dịch vụ tại Việt Nam
- 36/93 triệu người VN sd Internet (11/2014)

7

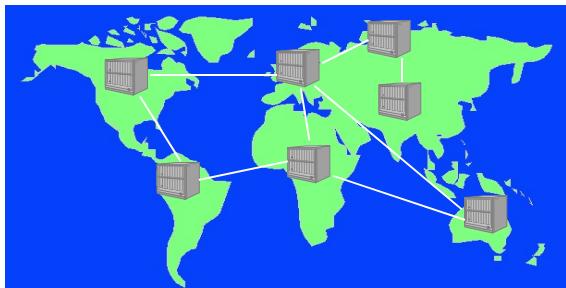
2. Một số khái niệm

- Internet là gì?
- Các giao thức SMTP, FTP, HTTP, URL.
- Web browser, Web server.
- Một số thuật ngữ thông dụng trong trang web.

8

Web – Mạng máy tính

Internet là một liên mạng máy tính toàn cầu được hình thành từ các mạng nhỏ hơn, liên kết hàng triệu máy tính trên thế giới thông qua cơ sở hạ tầng viễn thông. Internet là mạng của các mạng máy tính. Trong mạng này, các máy tính và thiết bị mạng giao tiếp với nhau bằng một ngôn ngữ thông nhất. Đó là bộ giao thức TCP/IP (Transmission Control Protocol – Internet Protocol).



9

Web- Mạng thông tin



10

❖ Web là gì ?

Web Là các dịch vụ phân tán cung cấp thông tin multimedia dựa trên hypertext

- **Phân tán:** Thông tin được đặt trên nhiều máy chủ ở khắp thế giới.
- **Multimedia:** Thông tin bao gồm text, graphics, sound, video
- **Hypertext:** Là kỹ thuật được sử dụng để truy cập thông tin

Cung cấp truy cập vào các tài nguyên mạng: FTP, News, ...

11

Các thành phần của trang web

- **Web Page**
 - Là một trang web
 - Có thể viết bằng các ngôn ngữ khác nhau nhưng kết quả trả về client là trang HTML
- **Web site**
 - Là tập hợp các trang Web có nội dung thống nhất, phục vụ cho một mục đích nào đó.
- **Uniform Resource Locator (URL):** Tham chiếu đến các tài nguyên trên Internet
<https://mail.google.com/mail/u/0/#inbox/>
- **Web Server, Web Browser**

12

- ❖ **URL-Uniform Resource Locator:**(Chuỗi định vị tài nguyên, là địa chỉ Web được xác định duy nhất)
 - **URL tuyệt đối** – là địa chỉ Internet đầy đủ của một trang hoặc file, bao gồm giao thức, vị trí mạng, đường dẫn tuy chọn và tên file. Ví dụ, <http://www.microsoft.com/ms.htm>.
 - **URL tương đối** - mô tả ngắn gọn địa chỉ tập tin kết nối có cùng đường dẫn với tập tin hiện hành, URL tương đối đơn giản bao gồm tên và phần mở rộng của tập tin.

13

- **Địa chỉ IP** (IP là viết tắt của từ: *Internet Protocol*) là một địa chỉ đơn nhất mà những thiết bị điện tử hiện nay đang sử dụng để nhận diện và liên lạc với nhau trên mạng máy tính bằng cách sử dụng giao thức Internet.

Một địa chỉ IP là một số 32-bit (IPv4) – 1995
IP 128 bit (IPv6) - 2000.

- **Tên Miền** (Domain Name)

Là tên giao dịch của công ty hay tổ chức trên Internet.

Ví dụ: www.cntp.edu.vn :

Là địa chỉ của 1 máy chủ thuộc tổ chức Trường ĐH CNTP
Có địa chỉ IP là 198.175.96.33, có tên máy chủ là WWW
Là tên miền cấp 1 (.com, .org, .edu, .biz, .net,)

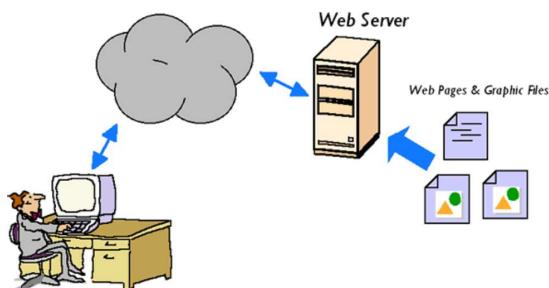
- **Ánh xạ tên miền và địa chỉ IP**

Do DNS server – Domain Name System (service) đảm trách

14

Web server

- Web Server là máy tính lớn được kết nối với tập hợp mạng máy tính mở rộng. Máy chủ có thể đọc các ngôn ngữ như file *.htm và *.html... → Máy chủ là kho để chứa toàn bộ dữ liệu hoạt động trên internet mà nó được giao quyền quản lý.
- Web server là một máy tính có dung lượng lớn, tốc độ rất cao để có thể lưu trữ vận hành tất cả dữ liệu trên internet. Các web server này phải đảm bảo hoạt động liên tục không ngừng nghỉ để duy trì cung cấp dữ liệu cho mạng lưới máy tính của mình.



15

Web Browser

- Lấy hiển thị (nếu có thể) các tài nguyên khác nhau
- Khả năng hiển thị:
Text-only (Lynx, ...)
Graphic (MSIE, Netscape, ...)
- Hiển thị được nhiều loại ảnh
TEXT, GIF, JPEG, sound, video, postscript, ...
- Hỗ trợ nhiều giao thức
HTTP, FTP, SMTP, POP, ...
- Có thể “plug-in” các công cụ vào browser để tăng tính năng
(3D animation, SWF, ...)

Web Browser : trình duyệt Web. Dùng để truy xuất các tài liệu trên các Web Server.

Internet Explorer, Google Chrome...

16

Các khái niệm khác

- **Mô hình Client-Server:** mô hình khách-chủ. Server chứa tài nguyên dùng chung cho nhiều máy Client
- **Internet Server:** là các Server cung cấp các dịch vụ Internet (Web Server, Mail Server, FTP Server...) Nestcape
- **Home page:** là trang web đầu tiên trong web site
- **Hosting provider:** là công ty hoặc tổ chức đưa các trang của chúng ta lên mạng
- **Publish:** Xuất bản trang web lên mạng
 - Hosting: thuê k/g lưu trữ trên máy chủ
 - Tên miền: thuê / mua tên miền từ nhà đăng ký tên miền
 - VD www.mozilla.org hoặc <http://www.bbc.co.uk>

17

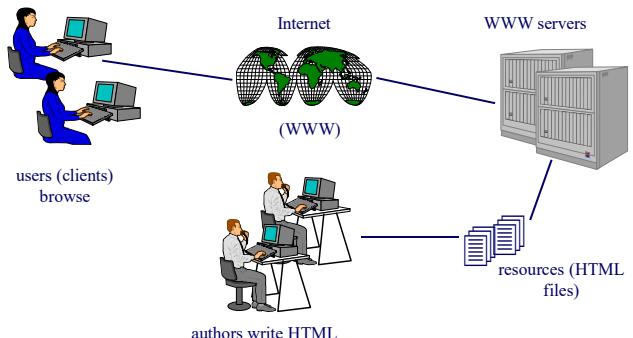
- **Internet Service Provider (ISP):** Nhà cung cấp dịch vụ Internet cho khách hàng. Mỗi ISP có nhiều khách hàng và có thể có nhiều loại dịch vụ Internet khác nhau.
- **Internet Protocol :** tiêu chuẩn chi phối việc chuyển tải thông tin giữa các máy tính trong mạng
- **World Wide Web (WWW):** dịch vụ tra cứu thông tin Internet. Dịch vụ này đưa ra cách truy xuất các tài liệu của các máy phục vụ dễ dàng thông qua các giao tiếp đồ họa. Để sử dụng dịch vụ này máy Client cần có một chương trình gọi là Web Browser.

18

- **Hyperlink:** siêu liên kết. Dùng để liên kết các trang web và dịch vụ của các website trên Internet.
- **IAP (Internet Access Provider):** Nhà cung cấp đường truyền Internet.
- **ISP (Internet Service Provider):** Nhà cung cấp dịch vụ Internet. Một số ISP hiện nay ở Việt Nam: VDC, FPT, SaigonNet, NetNam, ...

19

Cách thức hoạt động của Web



20

Web tĩnh – Web động

Trang web tĩnh

- Chứa nội dung cố định HTML
- Không cho phép NSD tương tác, cập nhật
- Một trang web chứa các hình ảnh chuyển động chưa hẳn là trang web động

Trang web động

- Kết hợp HTML và mã lệnh
- Mã lệnh được thực thi trên server, gửi kết quả là HTML về NSD
- Có khả năng tương tác với NSD

21

Ví dụ

Thiết kế một website giới thiệu gồm :

- Trang chủ
 - Logo, banner, Slogan
 - Danh sách thành viên
- Mỗi thành viên liên kết đến một trang cá nhân
 - Họ tên
 - Email
 - Hình
 - Thông tin khác

22

WEB TĨNH

Trang chủ
(Default.html)

Trang con
(member1.html)

Trang con
(member2.html)

...

23

WEB ĐỘNG

Trang chủ
(Default.aspx)

Trang con i
(member.aspx)

...

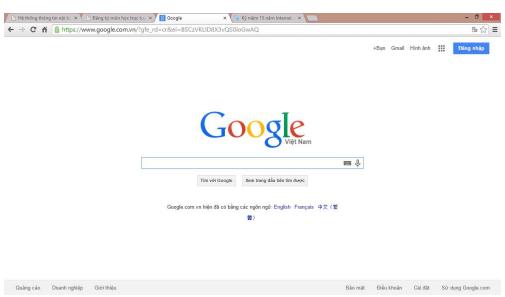
CSDL

Member1...
Member2...
Member3...
...

24

3. Các dịch vụ trên internet

- World Wide Web – WWW: là một hệ thống các máy chủ internet hỗ trợ các tài liệu định dạng bằng ngôn ngữ đánh dấu siêu văn bản



25

Thư điện tử – Email (Electronic mail)

- Là dịch vụ trao đổi các thông điệp qua mạng viễn thông
- Sử dụng giao thức SMTP/POP3 để gửi/nhận email
- Địa chỉ email có dạng: name@subdomain.domainname (VD: admin@cntp.edu.vn trong đó admin – Tên tài khoản email, cntp.edu.vn – tên miền)
- Được quản lý bởi Mail Server

Truyền, tải tập tin (FTP) – File Transfer Protocol

- Là dịch vụ trao đổi các tập tin giữa các máy tính trên Internet

Tán gẫu – Chat,...

26

Tra cứu thông tin

- www.google.com
- www.lycos.com
- www.altavista.com
- www.researchindex.com

(tra cứu bài báo khoa học)

Tin tức, thông tin tổng hợp



27

Nghiên cứu, khoa học, giáo dục

- www.codeproject.com
- <http://msdn.microsoft.com>
- www.programmersheaven.com

Mua bán trực tuyến

- Amazon (mua bán sách): www.amazon.com
- eBay (đấu giá): www.ebay.com
- Trang TMĐT



28

4. Khai thác tài nguyên

Download Website Offline

- Cho phép download website về và truy cập offline
- Teleport Pro – <http://www.tenmax.com>
- Offline Explorer Enterprise - <http://www.metaproducts.com>

Download File

- FlashGet – www.flashget.com
- GetRight - www.getright.com

Tìm kiếm

- Copernic – www.copernic.com: Hỗ trợ tìm kiếm thông minh trên nhiều Search Engine cùng lúc, và loại bỏ kết quả trùng

29

Chương 2.

HTML

1. Giới thiệu lịch sử
2. Các khái niệm căn bản
3. Các dịch vụ cơ bản của Internet
4. Khai thác tài nguyên trên Internet

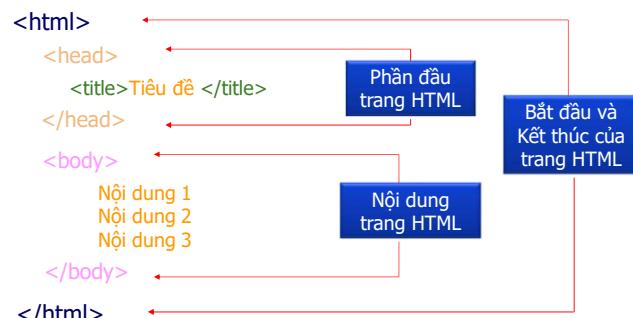
30

1. GIỚI THIỆU VỀ HTML

- ▶ HTML (Hyper Text Markup Language - Ngôn ngữ đánh dấu siêu văn bản)
 - ▶ Là một ngôn ngữ dùng để xây dựng một trang Web.
 - ▶ Chứa các thành phần định dạng để báo cho trình duyệt Web biết cách để hiển thị một trang Web.
 - ▶ Một trang web thông thường gồm có 2 thành phần chính:
 - Dữ liệu của trang web (văn bản, âm thanh, hình ảnh...)
 - Các thẻ (tag) HTML dùng để định dạng mô tả cách thức các dữ liệu ở trên hiển thị trên trình duyệt.
 - ▶ google “HTML tag b” hoặc <http://www.w3schools.com/html/DEFAULT.asp>
 - ▶ Các thẻ không phân biệt chữ hoa hay chữ thường
- HTML file có thể được tạo từ một trình **soạn thảo văn bản bất kỳ**

31

Cấu trúc trang HTML



32

Cấu trúc trang HTML

- `<html></html>` : Định nghĩa phạm vi của văn bản HTML
- `<head></head>` : Định nghĩa các mô tả về trang HTML. Thông tin trong tag này không được hiển thị trên trang web
- `<title></title>` : Mô tả tiêu đề trang web
- `<body></body>` : Xác định vùng thân của trang web, nơi chứa các thông tin

33

Cú pháp tag HTML

- ▶ `<Tag>` Dữ liệu `</Tag>`
- ▶ Tên Tag thường bắt nguồn từ một từ tiếng Anh
Ví dụ: B ~ Bold, I ~ Italic, P ~ Paragraph
Các thẻ không cần Tag đóng `
`, `<hr>`

Cú pháp :

`<tentag ThuocTinh1="GiaTri1" ThuocTinh2="GiaTri2" ...>`
Dữ liệu

`</tentag>`

VD:

`Danh sách có đánh số `
`<body bgcolor ="magenta"></body>`

34

HTML Style

- ❖ Sử dụng để thay đổi hiển thị cho các thành phần HTML đang tồn tại.
- ❖ Thuộc tính style có thể áp dụng cho bất kỳ thẻ HTML nào để hiệu chỉnh các dạng hiển thị color, background-color, text-align, font-family, font-size...
- ❖ Cú pháp của thuộc tính style
`style = "property:value"`

VD:

```
<p style="text-align:center; background-color:pink;">  
    This text is located at center side  
</p>
```

35

2. CÁC THẺ HTML

- Các Tag xử lý văn bản
- Tag hình ảnh
- Tag âm thanh
- Tag tạo danh sách
- Tag tạo liên kết
- Tag tạo bảng
- Form
- Frame

36

Các tag xử lý văn bản

Các thẻ định dạng khôi văn bản

- Tiêu đề (Heading) : `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>`, `<h6>`
- Đoạn văn bản (Paragraph): `<p>`
- Đường kẻ ngang (Horizontal Rules): `<hr>`

Các thẻ định dạng chuỗi văn bản

- Định dạng chữ: ``, `<i>`, `` và ``
- Xuống dòng: `
`

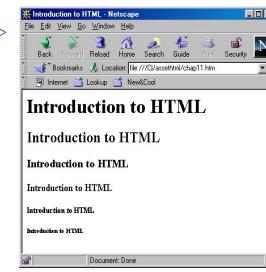
37

Các tag xử lý văn bản

Ví dụ về Heading

```
<HTML>
  <HEAD>
    <TITLE>Introduction to HTML</TITLE>
  </HEAD>

  <BODY>
    <H1>Introduction to HTML - H1</H1>
    <H2>Introduction to HTML - H2</H2>
    <H3>Introduction to HTML - H3</H3>
    <H4>Introduction to HTML - H4</H4>
    <H5>Introduction to HTML - H5</H5>
    <H6>Introduction to HTML - H6</H6>
  </BODY>
</HTML>
```



38

Try it Yourself » Try it Yourself » Try it Yourself »

st browsers will display the <h2>: st browsers will display the <h3>: st browsers will display the <h4>: st browsers will display the <h5>: st browsers will display the <h6> element with the following default values:

Example	Example	Example
<pre>h2 { display: block; font-size: 1.5em; margin-top: 0.83em; margin-bottom: 0.83em; margin-left: 0; margin-right: 0; font-weight: bold; }</pre>	<pre>h4 { display: block; font-size: 1em; margin-top: 1.33em; margin-bottom: 1.33em; margin-left: 0; margin-right: 0; font-weight: bold; }</pre>	<pre>h6 { display: block; font-size: .67em; margin-top: 2.33em; margin-bottom: 2.33em; margin-left: 0; margin-right: 0; font-weight: bold; }</pre>

39

Try it Yourself » Try it Yourself » Try it Yourself »

This is going to be real fun

Using another heading

Another paragraph element

```
<html>
  <head>
    <title>Welcome to HTML</title>
  </head>
  <body bgcolor='lavender'>
    <h3>My first HTML document</h3>
    <p>
      This is going to be real fun
      <h2>Using another heading</h2>
    </p>
    <p align='center'>Another paragraph element</p>
  </body>
</html>
```

40

Các tag xử lý văn bản

Ví dụ về <hr>

- `<HR ...>`
 - Thuộc tính :
 - align : Canh hàng đường kẻ ngang so với trang web
 - width : Chiều dài đường kẻ ngang
 - size : Bề rộng của đường kẻ ngang
 - noshade : Không có bóng
- `<HR noshade size='5' align='center' width='40%'></HR>`
- `<HR size='15' align='right' width='80%'></HR>`

41

Các tag xử lý văn bản

Tag định dạng chữ

```
<b>This text is bold</b>  
<strong>This text is strong </strong>  
<big>This text is big </big>  
<em>This text is emphasized </em>  
<i>This text is italic </i>  
<small>This text is small </small>  
This text contains a<sub>2</sub>  
This text contains x<sup>2</sup> = a x a
```

42

Các tag xử lý văn bản

Tag định dạng chữ

```
<EM>Computer Sciences</EM>  
<STRONG>Computer Sciences</STRONG>  
<DFN>Computer Sciences</DFN>  
<CODE>Computer Sciences</CODE>  
<KBD>Computer Sciences</KBD>  
<VAR>Computer Sciences</VAR>  
<CITE>Computer Sciences</CITE>  
<BLINK>Computer Sciences</BLINK>  
<DEL>Computer Sciences</DEL>  
<INS>Computer Sciences</INS>
```

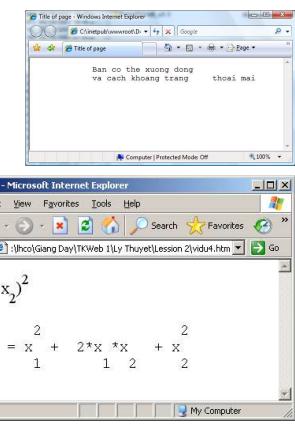
43

Các tag xử lý văn bản

Tag <pre>

Hiển thị đúng dạng văn bản đã soạn thảo
(khoảng trắng, xuống dòng, tag,...)

```
<pre>  
Ban co the xuong dong  
va cach khoang trang thoai mai  
</pre>  
  
<html>  
<head><title> vi du 4 </title></head>  
<body>  
(x<sub>1</sub>2 + x<sub>2</sub>2)<sup>2</sup>  
<pre>  
= x12 + 2*x1*x2 + x22  
</pre>  
</body>  
</html>
```



44

Các tag xử lý văn bản

Các ký tự đặc biệt ?

- Dấu <, >, & Dấu nháy kép “ Các ký tự đặc biệt : @ © ®

Ký tự	Mô tả	Tên	Mã
“ ”	Khoảng trống (non-breaking space)	 	
©	Bản quyền	©	©
®	Thương hiệu đã được đăng ký	®	®
™	Thương hiệu	™	™
€	Đồng Euro	€	€
£	Bảng Anh	£	£
<	Ký tự nhỏ hơn (less than)	<	<
>	Ký tự lớn hơn (greater than)	>	>
&	Ký hiệu & (ampersand)	&	&

45

Các tag xử lý văn bản

Định dạng màu sắc qua thuộc tính color

Màu sắc	Giá trị	Tên tiếng Anh
đỏ	#FF0000	RED
đỏ sẫm	#B00000	DARKRED
Xanh lá cây	#00FF00	GREEN
Xanh nhạt	#90EE90	LIGHTGREEN
Xanh nước biển	#0000FF	BLUE
Vàng	#FFFF00	YELLOW
Vàng nhạt	#FFFFFF	LIGHTYELLOW
Trắng	#000000	WHITE
Đen	#000080	BLACK
Xám	#A52A2A	GRAY
Nâu	#FF00FF	BROWN
Tím	#EE82EE	MAGENTA
Tím nhạt	#FFCCCB	VIOLET
Hồng	#FFA500	PINK
Da cam	#000080	ORANGE
Màu đồng phục hải quân	#4169E1	NAVY
	#7FFF00	ROYALBLUE
		AQUAMARINE

46

Ví dụ

```

<body>

<font color="#FF0000">Màu đỏ bằng số
</font><br>
<font color="red">Màu đỏ bằng chữ
</font><br>
<font color="green">
<h2> Mau GREEN </h2>
</font><br>
<font color="blue">Mau BLUE
</font><br>
<font color="#112233">Mau
&quot;112233&quot; </font><br>

</body>

```



47

Tag hình ảnh

Tag hình ảnh

- : Không có thẻ đóng
- Các thuộc tính của tag :
 - SRC : Đường dẫn đến file hình ảnh
 - ALT : Chú thích cho hình ảnh
 - Position: Top, Bottom, Middle
 - Border : Độ dày nét viền quanh ảnh (default=0)
- Đặt ảnh nền cho trang web
 - Sử dụng thẻ <body Background='Image Path'>

48

Tag âm thanh

Tag âm thanh <bgsound>

- <bgsound> : Không có thẻ đóng
 - Các thuộc tính của tag <bgsound>:
 - SRC : Đường dẫn đến file âm thanh
 - Loop : Số lần lặp (bằng -1 : Lặp vô hạn)
 - <bgsound> Thường đặt trong tag <head> của trang web Đặt ảnh nền cho trang web
- ❖ Ví dụ: <BGSOUND src='sound\ls1.mid' LOOP='1'>

49

Tag danh sách

Kiểu danh sách	Thẻ	Phản tử trong DS
Danh sách có thứ tự		
Danh sách không có thứ tự		
Danh sách tự định nghĩa	<DL>	<Dt>, <Dd>
Danh sách lồng nhau		
Dạng khác	<menu> <dir>	

50

Tag danh sách

- ▶ **Cú pháp:**

```
<OL Type=x Start=n>
<LI Type =x1 Value=m> Nội dung 1</li>
<LI Type =x1 Value=m> Nội dung 2</li>
...
</OL>
```
- ▶ **X: loại ký tự muốn sử dụng trong danh sách gồm :**
 - A: Chữ hoa
 - a: Chữ thường
 - I: Số la mã hoa
 - i: Số la mã thường
 - 1: Cho số mặc định

51

Tag danh sách

- ▶ n: giá trị đầu tiên của danh sách
- ▶ x1: là loại ký tự sử dụng cho dòng này và dòng tiếp theo, làm mất ảnh hưởng của x
- ▶ m: giá trị đầu tiên của dòng này, làm thay đổi giá trị của n

52

Tag danh sách

Ví dụ 1:

The screenshot shows a web browser window with the address bar pointing to 'D:\hco\Gia...'. The page content displays three types of ordered lists:

- numbered list:**
 1. apples
 2. bananas
 3. lemons
- letters list:**
 - a. apples
 - b. bananas
 - c. lemons
- roman numbers list:**
 - i. apples
 - ii. bananas
 - iii. lemons

The code for this example is:

```
<html>
<head>
<title>danh sach co thu tu </title></head>
<body>
<h3>numbered list:</h3>
<ol>
<li>1. apples</li>
<li>2. bananas</li>
<li>3. lemons</li>
</ol>
<h3>letters list:</h3>
<ol type="a">
<li>a. apples</li>
<li>b. bananas</li>
<li>c. lemons</li>
</ol>
<h3>roman numbers list:</h3>
<ol type="i">
<li>i. apples</li>
<li>ii. bananas</li>
<li>iii. lemons</li>
</ol>
</body>
</html>
```

53

Tag danh sách

Ví dụ 2:

The screenshot shows a web browser window with the address bar pointing to 'D:\hco\Gia...'. The page content includes several examples of ordered lists:

- Ordered lists can be very simple.**
 1. Item 1
 2. Item 2
 3. Item 3
- Ordered lists can have a variety of types.**
 - a. Lowercase letters
 - B. Uppercase letters
 - m. Lowercase Roman numerals
 - IV. Uppercase Roman numerals
 5. Arabic numerals
- Ordered lists can start at different values and with different types.**
 - j. This should be j
 - c. This should be c
 1. Lists can nest
 1. Nesting depth is unlimited

The code for this example is:

```
<html>
<head>
<title>Ordered List Example</title>
</head>
<body>
<p>Ordered lists can be very simple.</p>
<ol>
<li>1. Item 1</li>
<li>2. Item 2</li>
<li>3. Item 3</li>
</ol>
<p>Ordered lists can have a variety of types.</p>
<ol type="a">
<li>a. Lowercase letters</li>
<li>B. Uppercase letters</li>
<li>m. Lowercase Roman numerals</li>
<li>IV. Uppercase Roman numerals</li>
<li>5. Arabic numerals</li>
</ol>
<p>Ordered lists can start at different values and with different types.</p>
<ol type="j">
<li>j. This should be j</li>
<li>c. This should be c</li>
<li>1. Lists can nest<ol>
<li>1. Nesting depth is unlimited</li>
</ol>
</li>
</ol>
</body>
</html>
```

54

Tag danh sách

Cú pháp:

<UL Type= Shape1>
 <LI Type= Shape 2> Nội dung 1
 <LI Type= Shape 2> Nội dung 2
 ...

- *Shape 1, Shape 2:* là loại bullet tự động đặt ở đầu dòng trong danh sách
- *Shape 1:* ảnh hưởng đến toàn danh sách
- *Shape 2:* ảnh hưởng đến một mục trong danh sách

Các loại shape:

- *Circle:* Bullet tròn rỗng
- *Square:* Bullet vuông
- *Disc :* Bullet tròn đặc

55

Tag danh sách

Ví dụ 1:

The screenshot shows a web browser window with the address bar pointing to 'D:\hco\Gia...'. The page content displays three types of bullet lists:

- Disc bullets list:**
 - apples
 - bananas
 - lemons
- Circle bullets list:**
 - apples
 - bananas
 - lemons
- Square bullets list:**
 - apples
 - bananas
 - lemons

The code for this example is:

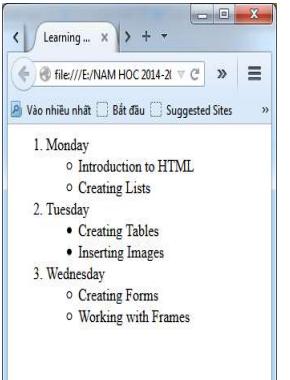
```
<html>
<head>
<title>danh sach khong thu tu </title></head>
<body>
<h3>Disc bullets list:</h3>
<ul type="disc">
<li>apples</li>
<li>bananas</li>
<li>lemons</li>
</ul>
<h3>Circle bullets list:</h3>
<ul type="circle">
<li>apples</li>
<li>bananas</li>
<li>lemons</li>
</ul>
<h3>Square bullets list:</h3>
<ul type="square">
<li>apples</li>
<li>bananas</li>
<li>lemons</li>
</ul>
</body>
</html>
```

56

Tag danh sách

- ▶ Ví dụ 2:

```
 1 <?xml version="1.0" encoding="UTF-8"?>
 2 <!DOCTYPE html>
 3 <html>
 4   <head>
 5     <title>Learning HTML</title>
 6     <meta http-equiv="Content-Type" content="text/html; charset=UTF-8;" />
 7   </head>
 8   <body>
 9     <h1>Monday</h1>
10     <ul>
11       <li><a href="#">Introduction to HTML</a></li>
12       <li><a href="#">Creating Lists</a></li>
13     </ul>
14     <h1>Tuesday</h1>
15     <ul type="disc">
16       <li><a href="#">Creating Tables</a></li>
17       <li><a href="#">Inserting Images</a></li>
18     </ul>
19     <h1>Wednesday</h1>
20     <ul type="circle">
21       <li><a href="#">Creating Forms</a></li>
22       <li><a href="#">Working with Frames</a></li>
23     </ul>
24   </body>
25 </html>
```



57

Tag tạo liên kết

- Thẻ <a> và các thuộc tính
 - Liên kết ngoại (external link)
 - Liên kết nội (internal link)
 - Bản đồ ảnh

Tag tạo liên kết

Thẻ <a> và các thuộc tính

- Cú pháp:

** nhᾶn **

- URL : địa chỉ của trang liên kết
 - Nhãn : có thể là text, hình ảnh hay button

59

Tag tạo liên kết

- ▶ Có 2 loại địa chỉ URL
 - Địa chỉ tuyệt đối
 - <A HREF =<protocol://host.domain:port/path/filename">Hypertext
 - / : vị trí thư mục gốc của website
 - Địa chỉ tương đối
 - Là vị trí tương đối so với trang web sử dụng link
 - .. : quay ra thư mục cha
 - VD: trong file A.HTM có hyperlink
 - Liên kết đến trang B
 - ▶ Liên kết email
 - Email

6

Tag tạo liên kết

- Thuộc tính TARGET của thẻ <A>

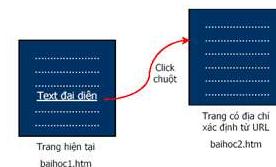
 hypertext

- name: Tải trang web vào frame có tên NAME
- _blank: Tải trang web vào cửa sổ mới
- _parent: Tải trang web vào cửa sổ cha của nó
- _self: Tải trang web vào chính cửa sổ hiện hành
- _top: Tải trang web vào cửa sổ cao nhất

61

Tag tạo liên kết

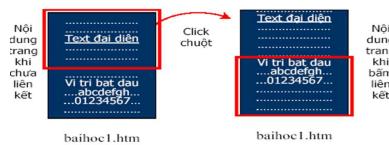
- Liên kết ngoại (external link): liên kết ra ngoài trang hiện hành



62

Tag tạo liên kết

- Liên kết nội (internal link): liên kết đến những phần trong cùng một trang web



63

Tag tạo liên kết

Nếu nội dung của trang quá dài thì nên tạo các Bookmark để khi xem, người dùng click vào bookmark để nhảy đến một phần cụ thể nào đó trên chính trang đó: gồm 2 bước

- Đánh dấu vị trí lk nội (Tạo BookMark):

 Tiêu đề Nội dung phân văn bản

- Tạo lk nội (Tạo liên kết đến Bookmark):

Nhãn của liên kết

Hoặc

Nhãn của liên kết

64

Tag tạo liên kết

Ví dụ

```
<HTML>
<HEAD><TITLE> Using html links</TITLE> </HEAD>
<BODY><font size=6>

<A HREF = '#Internet'>Internet</A><BR>
<A HREF = '#HTML'>Introduction to HTML</A><BR>

<A name = 'Internet'><b>Internet</b></A><br>
    Internet là một mạng của các mạng. Nghĩa là, các mạng máy tính được liên
    kết với các mạng khác, nối các nước và ngày nay là toàn cầu. Giao thức
    truyền thông là TCP/IP cung cấp liên kết với tất cả các máy tính trên thế
    giới<br>

<A name = 'HTML'><b>Introduction to HTML</b></A><BR>
    Ngôn ngữ đánh dấu siêu văn bản là ngôn ngữ chuẩn mà web sử dụng để tạo
    và nhận ra tài liệu. Mặc dù không phải là một tập con của ngôn ngữ nâng cấp
    tiêu chuẩn tổng quát (SGML), ngôn ngữ đánh dấu siêu văn bản cũng có liên
    quan với SGML. SGML là một phương pháp trình bày các ngôn ngữ đánh dấu
    tài liệu. HTML là ngôn ngữ đánh dấu được sử dụng để tạo tài liệu HTML. Các
    hướng dẫn chỉ rõ một trang web nên được hiển thị như thế nào trong trình
    duyệt
</font>
</BODY>
</HTML>
```

65

Tag tạo liên kết

Bản đồ ảnh

- **Bản đồ ảnh**
 - Bản đồ ảnh là một ảnh trong trang web được chia ra làm nhiều vùng, mỗi vùng khi click vào sẽ liên kết đến một địa chỉ URL
 - **Cách tạo:** Trước hết phải chèn vào trang một ảnh và đặt nhãn cho ảnh
 -
 - <Map Name="Label">
 - <Area Shape="type" coords="x1,y1,x2,y2, ..." href="URL">
 - <Area Shape="type" coords="x3,y3,x4,y4, ..." href="URL">
 - ...

66

Tag tạo liên kết

Bản đồ ảnh

- Bản đồ ảnh là một ảnh trong trang web được chia ra làm nhiều vùng, mỗi vùng khi click vào sẽ liên kết đến một địa chỉ URL
- **Cách tạo:** Trước hết phải chèn vào trang một ảnh và đặt nhãn cho ảnh
 -
 - <Map Name="Label">
 - <Area Shape="type" coords="x1,y1,x2,y2, ..." href="URL">
 - <Area Shape="type" coords="x3,y3,x4,y4, ..." href="URL">
 - ...

67

Tag tạo liên kết

Bản đồ ảnh

Trong đó:

- Label:** tên của bản đồ ảnh
- Type:** hình dạng của các vùng trên ảnh, gồm các loại:
 - Rect: Vùng hình chữ nhật
 - Circle: Vùng hình tròn
 - Poly: Vùng hình đa giác
- Coords:** tọa độ các đỉnh của hình
 - Rect: (x1, y1, x2, y2) là tọa độ 2 đỉnh chéo của vùng hình CN
 - Circle: (x, y, r) lần lượt là tọa độ tâm và bán kính của vùng hình tròn
 - Poly: (x1, y1, x2, y2, x3, y3, ...) là các đỉnh của vùng hình đa giác

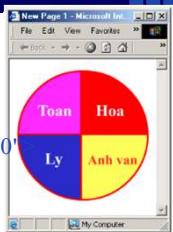
68

Tag tạo liên kết

• Ví dụ Bản đồ ảnh

```

<map name="Dalienket">
<area href="Toan.htm" shape="rect" coords="0, 0, 100, 100">
<area href="Ly.htm" shape="rect" coords="0,100, 100,200">
<area href="Hoa.htm" shape="rect" coords="100, 0,200, 100">
<area href="Anhvian.htm" shape="rect" coords="100,100,
200,200">
```



69

Tag tạo bảng

❖ Cấu trúc bảng tổng quát

```
<table>
<caption> TD Bảng </caption>
<tr>
    <th>Column I</th>
    <th>Column II</th>
</tr>
<tr>
    <td>Row 1, Col 1</td>
    <td>Row 1, Col 2</td>
</tr>
<tr>
    <td>Row 2, Col 1</td>
    <td>Row 2, Col 2</td>
</tr>
</table>
```

Thẻ	Ý nghĩa
<table>	Tạo bảng
<caption>	Tiêu đề mô tả bảng
<tr>	Tạo dòng
<th>	Tạo tiêu đề cột
<td>	Tạo cột

Column I	Column II
Row 1, Col 1	Row 1, Col 2
Row 2, Col 1	Row 2, Col 2

Kết quả trên trình duyệt
không có đường biên bao quanh

70

Tag tạo bảng

• Các thuộc tính thẻ <table>

Thuộc tính	Ý nghĩa
Bgcolor = "color"	Đặt màu nền cho bảng
Background ="URL"	Đặt hình nền
Border="n"	Độ dày đường viền
Bordercolor="color"	Màu đường viền
Cellpadding ="n"	Khoảng cách giữa nội dung và đường viền
Cellspacing="n"	Khoảng cách giữa các ô
Width="n" Height="n"	Kích thước bảng

71

Tag tạo bảng

• Các thuộc tính thẻ <tr>

Thuộc tính	Ý nghĩa
Bgcolor = "color"	Đặt màu nền cho dòng (hàng)
Bordercolor="color"	Màu đường viền cho dòng
Align ="center" "justify" "left" "right"	Canh lề theo phương ngang
Valign ="bottom" "middle" "top"	Canh lề theo phương đứng

72

Tag tạo bảng

- Các thuộc tính thẻ <td>, <th>

Thuộc tính	Ý nghĩa
Bgcolor = "color"	Đặt màu nền cho cell
Background ="URL"	Đặt hình nền cho cell
Bordercolor="color"	Màu đường viền
Align ="center" "justify" "left" "right"	Canh lề theo phương ngang
Valign ="bottom" "middle" "top"	Canh lề theo phương đứng
Rowspan="n" Colspan= "n"	Gộp nhiều dòng /cột lại thành một
Width="n" Height="n"	Kích thước cell

73

Tag tạo bảng

- Ví dụ:

Header 1	Header 2	Header 3
Row1, Col1	Row1, Col2	Row1, Col3
		Row2, Col2

```
<table border = 5 cellpadding = 20 cellspacing = 10 width = 600 height = 400>
<caption> VD2 </h2></caption>
<tr bcolor = blue align="center" valign= "middle">
    <th>Header 1</th>
    <th>Header 2</th>
    <th>Header 3</th>
</tr>

<tr align= "center" valign="top">
    <td>Row1, Col1</td>
    <td align="middle">Row1, Col2</td>
    <td align="bottom">Row1, Col3</td>
</tr>

<tr align="center" valign="middle">
    <td background= "hinh.jpg">Row2, Col1</td>
    <td bcolor= "#FFCC00" colspan=2>Row2, Col2</td>
</tr>
</table>
```

74

Form

- Giới thiệu :
- Cách tạo
- Các điều khiển

75

Form

- Giới thiệu

Form được sử dụng khi cần

- Thu thập thông tin tên, địa chỉ, số điện thoại, email, ...để đăng ký cho người dùng vào một dịch vụ, một sự kiện
 - Tập hợp thông tin để mua hàng
 - Thu thập thông tin phản hồi về một Website
- Form là các mẫu để user điền dữ liệu vào và gửi đến server

76

Form

- Cách tạo**

Cú pháp:

```

<Form Name = "Tên form"
      Method = "Post/ Get"
      Action = "link xử lý">
    Nội dung của Form
</Form>
```

77

Form

Trong đó:

- Method:** xác định phương thức đưa dữ liệu lên máy chủ, có 2 giá trị :Post và Get
 - Nếu giá trị là GET thì trình duyệt sẽ tạo một câu hỏi chứa trang URL, một dấu hỏi và các giá trị do biểu mẫu tạo ra. Trình duyệt sẽ đổi script của câu hỏi thành kiểu được xác định trong URL để xử lý.
 - Nếu giá trị là POST thì dữ liệu trên biểu mẫu sẽ được gửi đến SERVER như một khối dữ liệu
- Action:** là địa chỉ của script sẽ thực hiện khi form được submit (đường dẫn để nơi dữ liệu cần XL)

78

Form

- Các điều khiển**

Các phần tử của form thường sử dụng trên web gồm:

- **Input boxes:** nhập dữ liệu dạng text và number
- **Radio buttons:** dùng để chọn một tùy chọn trong danh sách
- **Selection lists:** dùng cho một danh sách dài các lựa chọn, thường là trong Drop-down list box
- **Check boxes:** chỉ định một item được chọn hay không
- **Text area:** một text box có thể chứa nhiều dòng
- **Submit và Reset button:** để gửi form đến url script vừa để reset form về trạng thái ban đầu

79

Form

INPUT TAG

Thẻ được sử dụng nhiều nhất để tạo các thành phần trên form là <input>

Thuộc tính **type** của <input> quyết định kiểu của thành phần

- ▶ **Cú pháp:**

```

<FORM>
  <INPUT TYPE="Object" NAME="Name">
</FORM>
```
- ▶ **Các giá trị của thuộc tính TYPE:**

Mặc định giá trị của TYPE là text, nếu trong tag <INPUT> không nhập thuộc tính TYPE thì loại input boxes là text

80

Form

- **Textbox:** là hộp văn bản do người SD nhập vào

‣ **Cú Pháp:**

```
<Input Type="Text" Value="Value"  
      Name="name" Size=n  
      Maxlength=m>
```

- Name : tên dữ liệu đầu vào server
- Value: Dữ liệu ban đầu có sẵn trong text box
- Size: chiều rộng của text box tính bằng số ký tự (mặc định là 20)
- Maxlength: số ký tự tối đa có thể nhập vào text box

81

Form

- **Hộp password:** Những ký tự nhập vào hiển thị dưới dạng dấu chấm , thông tin sẽ không bị mã hoá khi gửi lên server

‣ **Cú Pháp:**

```
<Input Type="Password" Name="name"  
      Size=n Maxlength=m>
```

- Name : tên dữ liệu đầu vào server
- Size: chiều rộng của hộp password tính bằng số ký tự
- Maxlength: số ký tự tối đa có thể nhập vào hộp password

82

Form

- **Checkbox:** Hộp chọn, người xem có thể đánh dấu nhiều checkbox trong cùng một bộ

‣ **Cú Pháp:**

```
<Input Type="checkbox" Name="name"  
      Value="Value" Checked > nhän
```

- Name: tên của checkbox
- Value: xác định mỗi giá trị cho mỗi hộp checkbox được gửi cho server khi người xem đánh dấu vào checkbox
- Checked: thuộc tính để hộp check box được chọn mặc định

83

Form

- **Radio button:** Cho phép người xem chỉ chọn một tùy chọn tại mỗi thời điểm

‣ **Cú Pháp:**

```
<Input Type="radio" Name="name"  
      Value="Value" Checked > nhän
```

- Name: tên của radio, kết nối các radio button với nhau
- Value: Những dữ liệu sẽ gửi đến server khi radio button được chọn
- Checked: thuộc tính để radio button được chọn mặc định

84

Form

- ▶ **Submit button:** Tất cả thông tin của người xem nhập vào sẽ được gửi đến server khi người xem click nút Submit

- ▶ **Cú Pháp:**

```
<Input Type="Submit" Name="name"  
Value ="Submit Message" >
```

- Submit Message: Là chữ xuất hiện trên Button
- Name: tên của button

85

Form

- ▶ **Reset button:** Thiết lập giá trị ban đầu của tất cả các điều khiển trên form

- ▶ **Cú Pháp:**

```
<Input Type="Reset" Name="name"  
Value ="Reset Message" >
```

- Reset Message: Là chữ xuất hiện trên Button
- Name: tên của button

86

Form

- ▶ **Button:** dùng để thực hiện các lệnh do người sử dụng đưa ra

- ▶ **Cú Pháp:**

```
<input type="button" name="Button"  
value="GỬI">
```

- Name: tên của button
- Value: Chữ sẽ hiện trên button

87

Form

- ▶ **Drop down menu**

- ▶ **Cú pháp:**

```
<Select Name="Name" size = "n" multiple>  
  <Option Value="Value" selected> Option 1  
  <Option Value="Value" > Option 2  
  ...  
</Select>
```

- Name: tên dữ liệu đầu vào server
- Size: là chiều cao của menu tính bằng hàng chữ
- Multiple: là thuộc tính cho phép chọn nhiều đề mục (listbox)
- Selected: đề mục được chọn mặc định
- Value: xác định dữ liệu gửi cho server nếu đề mục được chọn

88

Form

- ▶ **OptGroup :**
được sử dụng để nhóm các chọn lựa thành các nhóm riêng trong dropdown menu hoặc listbox
- ▶ **Cú pháp:**

```

<SELECT name= "name" >
    <OPTGROUP [ label="label"] [ Disable =
        "disable"]>
        <OPTION value= "value1">option1
        <OPTION value= "value2"> option2
        <OPTION value= "value3"> option3
        ...
    </OPTGROUP>
    ...
</SELECT>
```

89

Form

- **Ví dụ: OptGroup**



90

Form

- ▶ **Text area :**
Hộp văn bản cho phép nhập nhiều dòng
- ▶ **Cú pháp:**

```

<TextArea Name="name" Rows=n Cols=m
    readonly= "readonly" disabled = "yes">
    Default text
</textarea>
```

- Rows: số dòng có thể nhập vào TextArea (mặc định là 4)
- Cols: độ rộng của textarea, tính bằng số ký tự, mặc định
là 40

91

Framset

- Giới thiệu
- Frame dòng
- Frame cột
- Frame dòng cột kết hợp
- Các thuộc tính

92

Framset

- **Giới thiệu**

- Khi cần hiển thị trên trình duyệt nhiều nội dung, chủ đề khác nhau trên cùng một trang thì một giải pháp có thể đáp ứng cho trường hợp này là frame (khung).
- Có thể phân chia một trang thành các khung, cho phép người truy cập cùng một lúc có thể xem nhiều trang mà không cần cuốn màn hình, mỗi khung chứa một trang web riêng.
- Nếu trong trang đã sử dụng Frame thì không sử tag Body
- Là một trong các cách để bố cục cho trang.

93

Framset

- Cú pháp

```
<HTML>
  <HEAD>
    <TITLE>Page Title</TITLE>
  </HEAD>
  <FRAMESET>
    Frame Definitions
  </FRAMESET>
</HTML>
```

94

Framset

- **Frame dòng**

```
<HTML>
  <HEAD><TITLE>Nội dung tiêu đề</TITLE></HEAD>
  <Frameset Rows="a, b..." >
    <Frame name="Name1" Src="Content1.htm">
    <Frame name="Name2" Src="Content2.htm">
    ....
    <Frame name="Name_n" Src="Content_n.htm">
  </Frameset>
</HTML>
```

95

Framset

Trong đó:

- *a, b*: là độ cao của các dòng thứ 1, thứ 2 ..., có thể tính bằng pixel hoặc bằng %
- *Name*: tên khung, (xác định chức năng của khung)
- *Content.htm*: địa chỉ trang web xuất hiện đầu tiên trong khung

96

Framset

- Frame cột

Cú pháp:

```
<HTML>
<HEAD><TITLE>Nội dung tiêu đề</TITLE></HEAD>
<Frameset Cols="a, b...">
    <Frame name="Name1" Src="Content1.htm">
    <Frame name="Name2" Src="Content2.htm">
    ....
    <Frame name="Name_n" Src="Content_n.htm">
</Frameset>
</HTML>
```

97

Framset

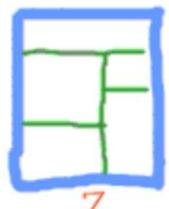
Trong đó:

- a, b: là độ rộng của các cột thứ 1, thứ 2 ..., có thể tính bằng pixel hoặc bằng %
- Name: tên khung, (xác định chức năng của khung)
- Content.htm: địa chỉ trang web xuất hiện đầu tiên trong khung

98

Framset

- Frame dòng cột kết hợp
 - Kết hợp dòng cột
 - Kết hợp cột dòng



```
<title> Bai 1 </title>      --
</head>
<frameset rows = "120,*">
    <frame src = "row1.htm">
    <frameset cols="75%,25%">
        <frameset rows="60%,40%">
            <frame src="Row2Col1Row1.html">
            <frame src="Row2Col1Row2.html">
        </frameset>
        <frameset rows="100,*">
            <frame src="Row2Col2Row1.html">
            <frame src="Row2Col2Row2.html">
        </frameset>
    </frameset>
</frameset>
```

99

Framset

- Các thuộc tính Frame

Noresize: Không đổi kích thước

Scrolling: có/không có thanh cuộn

Auto: Xuất hiện thanh cuộn khi nội dung dài

Yes: luôn xuất hiện thanh cuộn

No: không xuất hiện thanh cuộn

Ví dụ:

```
<frameset rows="80,*" frameborder="NO" border="0" >
    <frame noresize src="topFrame" scrolling="NO" >
    <frame noresize src="leftFrame" scrolling="NO" >
</frameset>
```

100

Framset

- Các thuộc tính Frame
 - Frameborder: đường viền của khung mặc định là 1, muốn giữa các khung không còn đường viền thì trong tag Frameset nhập thêm Border=0,
 - Marginwidth: hiệu chỉnh khoảng cách từ nội dung đến lề trái và phải của khung (tính bằng pixel)
 - Marginheight: hiệu chỉnh khoảng cách từ nội dung đến lề trên và dưới của khung (tính bằng pixel)

101

Framset

- Các thuộc tính Frame
 - Frameborder: đường viền của khung mặc định là 1, muốn giữa các khung không còn đường viền thì trong tag Frameset nhập thêm Border=0,
 - Marginwidth: hiệu chỉnh khoảng cách từ nội dung đến lề trái và phải của khung (tính bằng pixel)
 - Marginheight: hiệu chỉnh khoảng cách từ nội dung đến lề trên và dưới của khung (tính bằng pixel)

102

Chương 3.

CSS

- Giới thiệu CSS
- CSS Selector
- Phân loại CSS
- Một số thuộc tính định dạng CSS
- CSS – Background
- CSS – Font
- CSS – Text
- Box Model
- Pseudo class
- Float – Clear - Position

103

Giới thiệu

104

Giới thiệu

- ❖ Trong lĩnh vực xây dựng, chúng ta có trang trí nội thất; trong lĩnh vực thẩm mỹ - làm đẹp, chúng ta có kỹ thuật make-up; còn trong lĩnh vực thiết kế web chúng ta có CSS.
- ❖ CSS : Cascading Style Sheet là một ngôn ngữ quy định cách trình bày cho các tài liệu viết bằng HTML, XHTML, XML, SVG, hay UML
- ❖ CSS : là công cụ để trang trí web

TẠI SAO CẦN CSS ?

- ❖ Khi chúng ta cần thay đổi các định dạng (màu chữ, cỡ chữ , ...) → công việc nhảm chán → sử dụng CSS

❖ Ví dụ

CSS Selector

```
CSS Selector {  
    Property : value ;  
}
```

Trong đó: Css Selector có 4 loại

1. Html selector:

- Áp dụng cho các thành phần cần định dạng là thẻ html
- Css selector = tên thẻ html

```
<head>  
    <title>Bài 2. CSS Selector</title>  
    <style>  
        p{  
            color:red;  
            font-size:1.5em;  
            font-weight:bold;  
        }  
    </style>  
</head>
```

```
<body>  
    <p>Đây là đoạn văn</p>  
    <p>Đây là đoạn văn</p>  
    <p>Đây là đoạn văn</p>  
    <p>Đây là đoạn văn</p>  
    <p>Đây là đoạn văn</p>  
</body>
```

Đơn vị chiều dài			
Đơn vị	Mô tả	Đơn vị	Mô tả
%	Phần trăm	ex	1 ex bằng chiều cao của chữ x in thường của font hiện hành. Do đó, đơn vị này không những phụ thuộc trên kích cỡ font chữ mà còn phụ thuộc loại font chữ vì cùng 1 cỡ 14px nhưng chiều cao chữ x của font Times và font Tohama là khác nhau.
in	Inch (1 inch = 2.54 cm)	pt	Point (1 pt = 1/72 inch)
cm	Centimeter	pc	Pica (1 pc = 12 pt)
mm	Millimeter	px	Pixels (điểm ảnh trên màn hình máy tính)
em	1 em tương đương kích thước font hiện hành, nếu font hiện hành có kích cỡ 14px thì 1 em = 14 px. Đây là một đơn vị rất hữu ích trong việc hiển thị trang web.		

CSS Selector

❑ Ta có thể thực hiện gom nhóm các bộ chọn để thiết lập các tính chất chung cho nhiều bộ chọn.

❑ Ví dụ :

```

h1, h2, h3, h4, h5, h6 ← Các bộ chọn được phân cách nhau bằng dấu phẩy ","
{
  color: green ← Thiết lập thuộc tính color cho các bộ chọn có giá trị là green
}

```

CSS Selector

2. Class Selector: (bộ chọn lớp)

- Ví dụ xây một khu nhà gồm 20 căn có cấu trúc giống nhau, ta sẽ dùng bộ chọn lớp.
- **CSS_selector = .Ten_class**

```
.Ten_Class = { property : value;
}
```

```

p{
  color : red;
  font-size : 1.5em;
  font-weight : Bold;
}

</Style>
</head>
<body>

<p> Đây là đoạn văn bản !!! </p>
<Div> Đây là đoạn văn bản !!! </Div>
<Div> Đây là đoạn văn bản !!! </Div>
<p> Đây là đoạn văn bản !!! </p>

<style>
  .dinhdang {
    color:red;
    font-size:1.5em;
    font-weight:bold;
  }
</style>
</head>
<body>
  <p class="dinhdang">Đây là đoạn văn</p>
  <div class="dinhdang">Đây là đoạn văn</div>
  <p class="dinhdang">Đây là đoạn văn</p>
  <p class="dinhdang">Đây là đoạn văn</p>
  <p class="dinhdang">Đây là đoạn văn</p>

```

VD2:

Ví dụ: Trên trang web có 3 loại đoạn văn

- Đoạn văn canh lề trái
- Đoạn văn canh lề giữa
- Đoạn văn canh lề phải

Khi đó có thể định nghĩa 3 lớp riêng biệt (cho thẻ p) cho 3 loại đoạn văn này như sau:

```
p.trai {text-align: left}  
p.phai {text-align: right}  
p.giua {text-align: center}
```

VD3:

• Áp dụng vào trang html

```
<p class="trai">Đoạn văn này được canh lề trái.</p>  
<p class="phai">Đoạn văn này được canh lề phải.</p>  
<p class="giua">Đoạn văn này được canh lề giữa.</p>
```

• Áp dụng không hợp lệ

```
<p class="trai" class="phai">Đây là đoạn áp dụng sai</p>  
<td class="trai">Áp dụng sai thẻ</td>
```

VD4:

```
<style>  
p.one {  
    border-style: solid;  
    border-color: red;  
}  
  
p.two {  
    border-style: solid;  
    border-color: green;  
}  
  
p.three {  
    border-style: solid;  
    border-color: red green blue yellow;  
}  
</style>  
</head>  
<body>  
  
<h2>Thuộc tính màu của Border</h2>  
<p>Thuộc tính này chỉ định màu sắc của bốn đường viền:</p>  
  
<p class="one">A solid red border</p>  
<p class="two">A solid green border</p>  
<p class="three">A solid multicolor border</p>
```

tính màu của Border
này chỉ định màu sắc của bốn đường viền:
d border
een border
ulticolor border
"border-color" property does not work if it
ne. Use the "border-style" property to set
s first.

VD5:

• Ví dụ: định nghĩa một lớp "giua" có thể gắn với nhiều thẻ khác nhau:

```
.giua { text-align="center"; }
```

```
<p class="giua">Đoạn này canh lề giữa.</p>  
<td class="giua">Nội dung trong cột này được canh giữa.</td>
```

VD6:

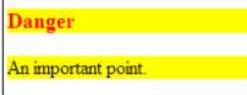
Một thẻ có thể được gán nhiều lớp bằng cách chỉ ra danh sách tên các lớp được phân cách nhau bằng dấu khoảng trắng.

Ví dụ:

```
.warning { color: red }
```

```
.highlight { background-color: yellow }
```

```
<h3 class="warning highlight">Danger</h3>  
<p class="highlight">An important point.</p>
```



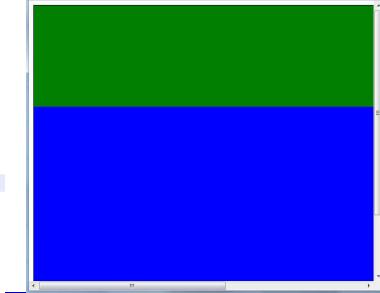
CSS Selector

3. ID Selector: (bộ chọn mã)

- Ví dụ xây một khu nhà gồm 2 căn có cấu trúc giống nhau
- Áp dụng với trường hợp là duy nhất
- CSS_selector = #Ten_ID
- #Ten_ID { property : value; }

CSS Selector

```
<style>#Wrapper { width: 960px; margin: auto; border: 1px Solid Black; height: 600px; }#Tren { height: 30%; background: Green; }#Duo1 { height: 70%; background: Blue; }</style></head><body> <div id = "Wrapper"> <div id = "Tren"> </div> <div id = "Duo1"> </div> </div></body>
```



CSS Selector

VD2: Khi ID không được gắn với Thẻ nào thì nó chỉ có tác dụng duy nhất cho 1 thẻ gọi nó đầu tiên

```
#xyz {color: red}
```

Đoạn văn bản có hiệu lực

Đoạn văn bản không có hiệu lực

Khi sử dụng:

```
<p id="xyz">Đoạn văn bản có hiệu lực </p>
```

```
<b id="xyz">Đoạn văn bản không có hiệu lực </b>
```

VD3: Dùng để chọn một thành phần / Thẻ nằm bên trong một thành phần / Thẻ khác

Ví dụ: Chỉ áp dụng style cho thẻ p nằm trong thẻ div

```
<html>
<head>
    <style type="text/css">
        div p {color:red; font-weight:bold;}
    </style>
</head>
<body>
    <div> <p>Đoạn văn bản có hiệu lực </p>
    <b><p>paragraph </p></b>
    </div>
    <p>Đoạn văn bản không có hiệu lực </p>
</body>
</html>
```

Đoạn văn bản có hiệu lực
Đoạn văn bản không có hiệu lực

CSS Selector

4. Child Selector: (bộ chọn con)

- Dùng để chọn một thành phần / Thẻ con của một thành phần / Thẻ khác
- CSS_selector = thẻ > thẻ > ... { property : value; }

Ví dụ 1: Chọn các thẻ p là con của thẻ div

```
div > p { color: red; }
```

Ví dụ 2: Chọn các thẻ b là con của thẻ p và thẻ p này lại là thẻ con của thẻ div

```
div > p > b { color: red; }
```

Phân loại CSS

1. CSS nội tuyến (nhúng vào từng thẻ html muốn áp dụng qua thuộc tính style)

- Ví dụ:

```
<html>
<head>
    <title>Ví dụ</title>
</head>
    <body style="background-color : #FFF;">
        <p style="color:green"> Welcome </p>
        <p> WERT </p>
    </body>
</html>
```

Phân loại CSS

2. CSS trong (dùng thẻ style)

- **Ví dụ:**Cũng ví dụ làm trang web có màu nền vàng, đoạn văn bản chữ xanh lá

```
<html>
<head>
    <title>Ví dụ</title>
    <style type="text/css">
        body { background-color:#FFF }
        p { color:#00FF00 }
    </style>
</head>
<body>
    <p> Welcome </p>
    <p> ABCDEF </p>
</body>
```

Phân loại CSS

3. CSS ngoài (liên kết với một file CSS bên ngoài) → thực tế sử dụng nhiều

- Ví dụ: Cũng ví dụ làm trang web có màu nền trắng, đoạn văn bản chữ xanh lá
- ```
<html>
<head>
 <title>Ví dụ</title>
 <link rel="stylesheet" type="text/css"
 href="style.css" />
</head>
<body>
 <p> Welcome </p>
</body>
</html>
```

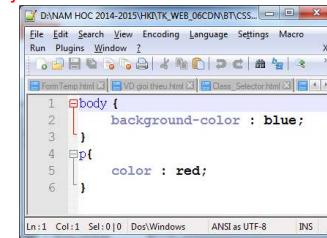
### Trong file “style.Css” viết gì ?

Style.css được tạo cùng cấp với file css Ngoai.html  
Css selector {

    Property : value;

}

### VÍ DỤ :



```
D:\NAM HỌC 2014-2015\HKT\TK_WB_06CDN\BT\CSS...
File Edit Search View Encoding Language Settings Macro
Run Plugins Window 2
FormTemp.html [] VD già thiệu.html [] Class Selector.html [] 4 | 2 |
1 body {
2 background-color : blue;
3 }
4 p{
5 color : red;
6 }
```

## SỰ ƯU TIÊN CÁC LOẠI CSS ?

Trình duyệt sẽ đọc tất cả các CSS được áp dụng cho trang ( CSS mặc định, file CSS ngoài, CSS nhúng trong thẻ, CSS nội tuyến)

### ❖ Thứ tự ưu tiên:

CSS nội tuyến → CSS trong → CSS ngoài →  
CSS mặc định của trình duyệt

### ❖ VD : Trong một trang web có liên kết tới file style.css có nội dung như sau:

```
p {
 color:#333;
 text-align:left;
 width:500px
}
```

trong thẻ <style> giữa thẻ <head> cũng có một đoạn CSS liên quan:

```
p {
 background-color:#FF00FF;
 text-align:right;
 width:100%;
 height:150px
}
```

trong phần nội dung trang web đó cũng có sử dụng CSS nội tuyến:

```
<p style="height:200px; text-align:center; border:1px solid #FF0000; color:#000" >
```

Vậy khi duyệt web thì CSS của trình duyệt sẽ đọc tất cả các CSS lưu vào một CSS ảo, nếu có sự trùng lắp của các thuộc tính thì nó lấy thuộc tính có độ ưu tiên cao hơn → CSS cuối cùng mà phần tử thẻ <p> nhận được là :

```
p {
 background-color:#FF00FF;
 width:100%;
 height:200px;
 text-align:center;
 border:1px solid #FF0000;
 color:#000
}
```

Trường hợp muốn thay đổi độ ưu tiên thì sử dụng thuộc tính **!important**

```
CP : CSS_Selector {
 property : value !important
}
```

VD:

```
p {
 width:500px;
 text-align:left !important;
 color:#333 !important
}

p {
 background-color:#FF00FF;
 width:100%;
 height:150px !important;
 text-align:right;
}

<p style="text-align:center; height:200px; border:1px solid #FF0000; color:#000" >

p {
 background-color:#FF0000;
 width:100%;
```

## CÁC THUỘC TÍNH

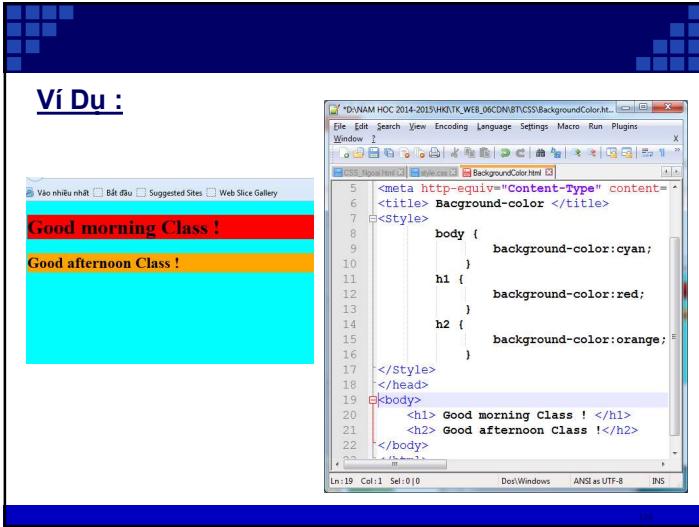
**1. Background-color:** định màu nền cho đ/tượng

VD :

```
<style>
 body {
 background-color : blue;
 }
</style>
....
```

```
<body bgcolor =“blue” >
</body>
```

Tham khảo bảng mã màu



**2. Background-image : Định hình nền cho trang web**

- ❖ VD :

```
<style>
 body {
 background-image : url("");
 }
</style>
```

- ❖ Trong TH kích thước ảnh quá nhỏ → trình duyệt mặc định sẽ lặp lại ảnh để phủ kín không gian còn thừa → để điều khiển việc lặp ta có thuộc tính **background-repeat**

- + Repeat-x : lặp ảnh theo phương ngang
- + Repeat-y : lặp ảnh theo phương dọc
- + Repeat : lặp ảnh theo cả hai phương (mặc định)
- + No-repeat : không lặp ảnh
- ❖ **Khóa ảnh nền:** **background-attachment** → xác định tính cố định của ảnh nền so với nội dung trang web.
  - + Ảnh nền cuộn cùng nội dung trang Web: **Scroll**
  - + Ảnh nền đứng yên so với nội dung trang web: **fixed**

❖ **Định vị ảnh nền:** `background-position` → xác định vị trí tọa độ của ảnh nền trên trang web.

Ví dụ :

Giá trị	Ý nghĩa
<code>Background-position:5cm 2cm</code>	Ảnh được định vị 5cm từ trái qua và 2cm từ trên xuống.
<code>Background-position:20% 30%</code>	Ảnh được định vị 20% từ trái qua và 30% từ trên xuống.
<code>Background-position:bottom left</code>	Ảnh được định vị ở góc trái phía dưới

**Thuộc tính background rút gọn :**

**VD:**

```
background-color:transparent;
background-image: url(logo.png);
background-repeat: no-repeat;
background-attachment: fixed;
background-position: right bottom;
```

thành một dòng ngắn gọn:

```
background: transparent url(logo.png) no-repeat fixed right bottom;
```

❖ Cấu trúc rút gọn của nhóm background

```
Background :<background-color> |
 | <background-image> |
 | <background-repeat> |
 | <background-attachment> |
 | <background-position> |
```

## CSS - FONT

1. **Font-family:** định font chữ trong trang web

**CSS Selector {**

```
Font-family : tênFont, tênFont2,...;
}
```

**VD:**

```
body { font-family:"Times New Roman",Tohama,sans-serif }
h1, h2, h3 { font-family:arial,verdana,serif }
```

→ Chú ý : đối với font chữ có khoảng trắng thì phải sử dụng cặp dấu “ ”

## CSS - FONT

**2. Font-style:** định dạng dáng chữ ( normal, italic, Oblique)

**VD:** <Style>

```
P { font-style: italic;
}
</Style>
<body>
 <p>CSS-Font</p>
</body>
```

## CSS - FONT

**3. Font-variant:** định dạng chữ giữ chế độ bình thường và small caps của một font chữ

**VD:**

```
<Style>
 h1 {
 font-variant:small-caps
 }
</Style>
```

## CSS - FONT

**4. Font-weight:** dùng để mô tả font chữ sử dụng in thường (normal) hay in đậm (Bold)

**VD:**

```
<Style>
 p {
 font-weight: bold;
 }
</Style>
```

## CSS - FONT

**5. Font-size :** dùng để xác định kích thước của một font chữ

**VD:**

```
body {
 font-size: 20px;
}
H1 {
 font-size : 3em;
}
```

## CSS - FONT

### 6. Thuộc tính Font rút gọn

Cấu trúc rút gọn cho các thuộc tính nhóm font:

**Font :** `<font-style>`

- | `<font-variant>` |
- | `<font-weight>` |
- | `<font-size>` |
- | `<font-family>`

## Ví dụ

```

h1 {
 font-style: italic;
 font-variant: small-caps;
 font-weight: bold;
 font-size: 35px;
 font-family: arial, verdana, sans-serif;
}

```

Thành

```

h1 {
 font: italic bold 35px arial, verdana, sans-serif;
}

```

Có thể vào [Font.google.com](https://font.google.com/specimen/Lemon) để sử dụng nhiều font-family trực tuyến

147

## CSS - TEXT

### 1. Thuộc tính color : định màu chữ

**VD:**

```

<style>
p{
 color : red;
}
</style>

```

```

<p style = "color : blue ; text-indent : 50px">

```

## CSS - TEXT

2. **Thuộc tính text – indent :** tạo khoảng lùi đầu dòng cho một đoạn text

**VD:**

```
<style>
 p{
 color : red;
 }
</style>

<p style = "color : blue ; text-indent : 50px">
```

## CSS - TEXT

3. **Thuộc tính text – align :** canh vị trí cho một đoạn văn bản (left, right, center,justify)

**VD:**

```
<style>
 p{
 text-indent : 50px;
 text-align : left;
 }
</style>
```

## CSS - TEXT

4. **Thuộc tính text – transform :**

- Uppercase: chữ hoa
- Lowercase : chữ thường
- Capitalize : viết hoa chữ đầu
- Overline : chữ gạch trên đầu
- Blink : chữ nhấp nháy.

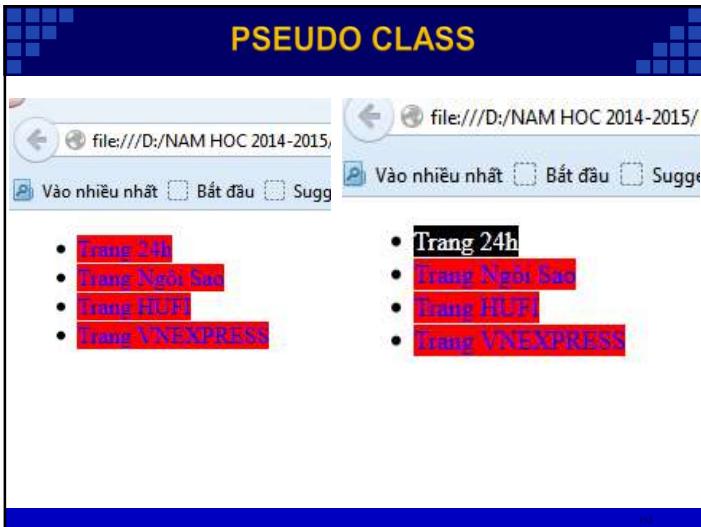
5. **Thuộc tính letter- spacing:**

## CSS - TEXT

6. **Thuộc tính word-spacing:**

7. **Thuộc tính text – decoration :**

- Underline
- Line-through
- None



## PSEUDO CLASS

### 1. Pseudo là gì ?

- Pseudo là phần tử giả
- Chức năng : tạo các hiệu ứng cho liên kết

### 2. Các loại Pseudo:

- Link : tạo hiệu ứng với các liên kết chưa được thăm
- Visited : liên kết đã xem
- Hover : di chuột qua liên kết
- Active : liên kết đang hoạt động (ấn giữ chuột)

### Ví dụ 1:

Tạo một danh sách các liên kết đến các trang như hình dưới đây

**Y/C:**

1. Các liên kết không có gạch chân, có background màu đỏ.
2. Các liên kết đã thăm có text màu xanh blue
3. Các liên kết khi hover chuột qua có text màu trắng, background màu đen
4. Các liên kết khi active có background màu xanh green

### Ví dụ 2:

Tạo các hiệu ứng tương ứng với tình trạng liên kết:

1. các liên chưa thăm có màu xanh lá, kích cỡ font 14px;
2. liên kết mouse hover có màu đỏ tươi, kích cỡ font 1.2em, hiệu ứng nhấp nháy;
3. liên kết đã thăm sẽ có màu xanh da trời, không có đường gạch chân;
4. các liên kết đang kích hoạt có màu tím và font dạng small-caps.

## Ví dụ 2:

```
a:link {
 color:#00FF00;
 font-size:14px
}
a:hover {
 color:#FF00FF;
 font-size:1.2em;
 text-decoration:blink
}
```

```
a:visited { color:#FF0000;
text-decoration:none
}
a:active {
color:# 662D91;
font-variant:small-caps
}
```

## BOX MODEL

### 1. Box Model là gì ?

- Box Model là : mô hình hộp, tạo các định dạng khối giữa các thành phần với nhau
- Cách mà CSS định dạng khối không gian bao quanh một thành phần nào đó

### 2. Các thành phần Box Model:

- Border ( viền)
- Padding ( vùng đệm)
- Margin (canh lề)



## BOX MODEL

### ❖ Thuộc tính border

- Được dùng trong trang trí, đóng khung cho một đối tượng cần nhấn mạnh, phân cách các đối tượng giúp trang web dễ nhìn hơn
- Cú pháp viết gọn :

Border : <border width>|

<border color>|

<border style>

## BOX MODEL

<border -width> : quy định độ rộng cho viền ( thin, medium, thick, hay một giá trị do cụ thể bằng pixel)

<border -color>: quy định màu viền cho một đối tượng web

<border -style>: quy định kiểu viền, có 8 kiểu đường viền ( solid, dotted, dashed, double, groove, inset, outset,...)



## BOX MODEL

### ❖ Thuộc tính margin :

- Được dùng để canh lè cho cả trang web hay một thành phần web, có 4 thành phần margin ( margin-top , margin-bottom, margin-left, margin-right )
- Cú pháp viết gọn :

Margin : <margin-top> |<margin- right>  
<margin- bottom>| <margin-left>

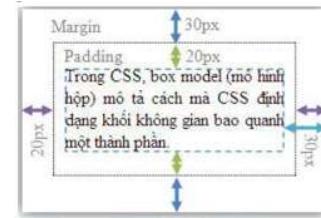
## BOX MODEL

- Hoặc cú pháp :

Margin : <value1>|<value 2>

Value1 : giá trị margin – top và margin- bottom

Value2: giá trị margin- left và margin - right



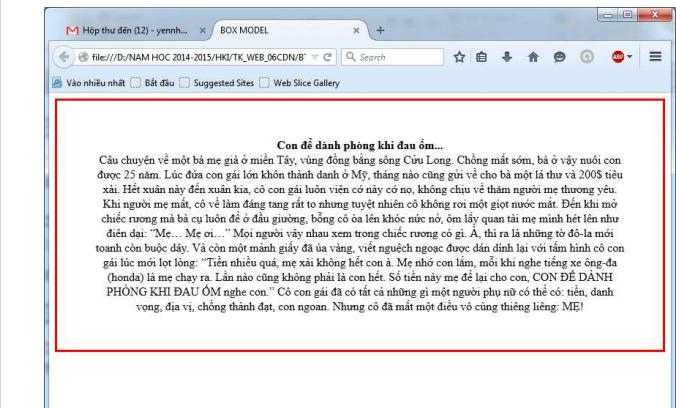
## BOX MODEL

**❖ Thuộc tính padding :**

- Quy định khoảng cách giữa phần nội dung và viền của một đối tượng. Có 4 yếu tố (*padding-top, padding-right, padding-bottom, padding-left*)
- Cú pháp viết gọn :

`padding : <padding-top> |<padding-right>  
                  <padding-bottom>|<padding-left>`

## Ví dụ:



**Con đẻ dành phòng khi đau ốm..**

Câu chuyện về một bà mẹ già ở miền Tây, vùng đồng bằng sông Cửu Long. Chồng mất sớm, bà ở vây nuôi con được 25 năm. Lúc đứa con gái lớn khôn thành danh ở Mỹ, tháng nào cũng gửi vé cho bà một lá thư và 200\$ tiêu xài. Hết xuân này đến xuân kia, cô con gái luôn viên có này cõi nõ, không chịu về thăm người mẹ thương yêu. Khi người mẹ mất, cô vẫn làm đám tang rất to nhưng tuyệt nhiên cô không rơi một giọt nước mắt. Đến khi mở chiếc rương mà bà cụ luôn để ở đầu giường, bỗng cô ôm khóc nức nở, ôm lấy quan tài mẹ mình hét lên như điên dại: "Mẹ... Mẹơi..." Mọi người vây nhau xem trong chiếc rương có gì. À, thi là những tờ đồ-là mới toanh còn buộc dây. Vâc còn một mảnh giấy đã úa vàng, viết nguêch ngoạc được dán dính lại với tấm hình cô con gái lõi mới lột lông: "Tiễn nhiều quá, mẹ xá không hết con à. Mẹ nhớ con lắm, mỗi khi nghe tiếng xe ông-đa (honda) là mẹ chạy ra. Lần nào cũng không phải là con hết. Số tiền này mẹ để lại cho con. CON ĐỀ DANH PHÒNG KHÍ ĐAU ỐM nghe con." Cô con gái đã có tất cả những gì một người phụ nữ có thể có: tiền, danh vong, địa vị, chứng thành đạt, con ngoan. Nhưng cô đã mất một điều vô cùng thiêng liêng: MẸ!

## FLOAT và CLEAR

**❖ Thuộc tính Float :**

- Dùng để cố định một thành phần của website
- Thuộc tính float có 3 giá trị
  - Left : cố định thành phần về bên trái
  - Right : cố định thành phần về bên phải
  - None : chế độ bình thường

## Ví dụ

Tạo 1  
trang gồm  
hai thành  
phàn như  
hình sau



**Con đẻ dành phòng khi đau ốm..**

Câu chuyện về một bà mẹ già ở miền Tây, vùng đồng bằng sông Cửu Long. Chồng mất sớm, bà ở vây nuôi con được 25 năm. Lúc đứa con gái lớn khôn thành danh ở Mỹ, tháng nào cũng gửi vé cho bà một lá thư và 200\$ tiêu xài. Hết xuân này đến xuân kia, cô con gái luôn viên có này cõi nõ, không chịu về thăm người mẹ thương yêu. Khi người mẹ mất, cô vẫn làm đám tang rất to nhưng tuyệt nhiên cô không rơi một giọt nước mắt. Đến khi mở chiếc rương mà bà cụ luôn để ở đầu giường, bỗng cô ôm khóc nức nở, ôm lấy quan tài mẹ mình hét lên như điên dại: "Mẹ... Mẹơi..." Mọi người vây nhau xem trong chiếc rương có gì. À, thi là những tờ đồ-là mới toanh còn buộc dây. Vâc còn một mảnh giấy đã úa vàng, viết nguêch ngoạc được dán dính lại với tấm hình cô con gái lõi mới lột lông: "Tiễn nhiều quá, mẹ xá không hết con à. Mẹ nhớ con lắm, mỗi khi nghe tiếng xe ông-đa (honda) là mẹ chạy ra. Lần nào cũng không phải là con hết. Số tiền này mẹ để lại cho con. CON ĐỀ DANH PHÒNG KHÍ ĐAU ỐM nghe con." Cô con gái đã có tất cả những gì một người phụ nữ có thể có: tiền, danh vong, địa vị, chứng thành đạt, con ngoan. Nhưng cô đã mất một điều vô cùng thiêng liêng: MẸ!

## FLOAT và CLEAR

### ❖ Thuộc tính Clear :

- Clear là thuộc tính đi cùng thuộc tính Float
- Clear là thuộc tính được gắn vào các phần tử liên quan đến phần tử đã được float để định hướng xử lý cho phần tử này.
  - Left : tràn bên trái
  - Right : tràn bên phải
  - Both : không tràn
  - None : chế độ bình thường

## POSITION

- Position dùng để định vị trí của phần tử
- Position gồm 2 yếu tố
  - Relative: định vị trí tương đối của phần tử với thành phần mẹ.
  - Absolute :định vị trí tuyệt đối của phần tử, có thể đặt ở bất cứ đâu trên trang web.

## Ví dụ

```
#logo1 { position:absolute; top:50px; left:70px }
#logo2 { position:absolute; top:0; right:0 }
#logo3 { position:absolute; bottom:0; left:0 }
#logo4 {position:absolute; bottom:70px; right:50px }
```

## HTML CANVAS

- **Canvas** là một phần tử của HTML5, cho phép thực hiện lập trình kết xuất đồ họa các đối tượng hai chiều trên trang web.

- **Cú pháp:**

```
<Canvas ID = "Example"
Width = "500"
Height = "300">
```

Đoạn văn bản này hiển thị vì trình duyệt của bạn không hỗ trợ

```
</Canvas>
```

172

## HTML CANVAS

- **Canvas** chiếm một khu vực trong trang, sau đó mã Javascript có thể truy cập vào khu vực này để vẽ thông qua một tập các hàm đồ họa tương tự như các API 2D, từ đó cho phép đồ họa động. Canvas có thể dùng để vẽ các đồ thị, hoạt họa, làm trò chơi, sang tác hình ảnh,...
- **Các Web browser hỗ trợ**

Element	Chrome	Edge	Firefox	Opera	Safari
<canvas>	4.0	9.0	2.0	3.1	9.0

173

## HTML CANVAS

- **Ví dụ 1: Draw a Line**



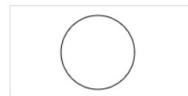
```
<canvas id="myCanvas" width="200" height="100" style="border:1px solid #000000;">
</canvas>
```

```
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.moveTo(0, 0);
ctx.lineTo(200, 100);
ctx.stroke();
</script>
```

174

## HTML CANVAS

- **Ví dụ 2: Draw a Circle**



```
<canvas id="myCanvas" width="200" height="100" style="border:1px solid #000000;">
</canvas>
```

```
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.beginPath();
ctx.arc(95, 50, 40, 0, 2 * Math.PI);
ctx.stroke();
</script>
```

175

## HTML CANVAS

- **Ví dụ 3: Draw a Text**

Hello World

```
<canvas id="myCanvas" width="200" height="100" style="border:1px solid #000000;">
</canvas>
```

```
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.font = "30px Arial";
ctx.fillText("Hello World", 10, 50);
</script>
```

176

## HTML CANVAS

- **Ví dụ 4:** Draw a Stroke Text



```
<canvas id="myCanvas" width="200" height="100" style="border:1px solid #000000;">
</canvas>
```

```
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.font = "30px Arial";
ctx.strokeText("Hello World", 10, 50);
</script>
```

177

## HTML CANVAS

- **Ví dụ 5:** Draw Linear Gradient



```
<canvas id="myCanvas" width="200" height="100" style="border:1px solid #000000;">
</canvas>
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");

// Create gradient
var grd = ctx.createLinearGradient(0, 0, 200, 0);
grd.addColorStop(0, "red");
grd.addColorStop(1, "white");

// Fill with gradient
ctx.fillStyle = grd;
ctx.fillRect(10, 10, 150, 80);
</script>
```

178

## HTML CANVAS

- **Ví dụ 6:** Draw Circular Gradient



```
<canvas id="myCanvas" width="200" height="100" style="border:1px solid #000000;">
</canvas>
```

```
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");

// Create gradient
var grd = ctx.createRadialGradient(75, 50, 5, 90, 60, 100);
grd.addColorStop(0, "red");
grd.addColorStop(1, "white");

// Fill with gradient
ctx.fillStyle = grd;
ctx.fillRect(10, 10, 150, 80);
</script>
```

179

## HTML CANVAS

- **Ví dụ 7:** Draw Image

Image to use:



Canvas to fill:



TRY IT

```
<!DOCTYPE html>
<html>
<body>

<p>Image to use:</p>

<p>Canvas to fill:</p>
<canvas id="myCanvas" width="250" height="300"
style="border:1px solid #3d3d3d;">
Your browser does not support the HTML canvas tag.</canvas>

<p><button onclick="myCanvas()">Try it</button></p>

<script>
function myCanvas() {
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
var img = document.getElementById("scream");
ctx.drawImage(img,10,10);
}
</script>

</body>
</html>
```

180

## Display

Các Element trong HTML có các giá trị hiển thị mặc định **block** hay **inline**

- Một block –level Element luôn bắt đầu trên một dòng mới và full hết chiều rộng dòng.

VD: <div>; <h1> - <h6>; <p>; <form>; <header>; <footer>; <section>

- Một inline level Element: không bắt đầu một dòng mới và chỉ nhận độ rộng vừa với nội dung

VD: <span>; <a>; <img>

181

## Display

**Display:none;** → ẩn một Element mà không phải xóa chúng và trang sẽ được hiển thị như là element không tồn tại.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>This is a visible heading</title>
</head>
<body>
<h1>This is a visible heading</h1>
<!-- This is a comment -->
<p>Notice that the h1 element with display: none; does not take up any space.</p>
</body>
</html>
```

This is a visible heading  
Notice that the h1 element with display: none; does not take up any space.

**Visibility: hidden;** → cũng cho phép ẩn Element, tuy nhiên Element vẫn giữ cùng khoảng cách trước (still take up the same space as before).

182

## Display

**Display:block;** → cho phép hiển thị một Element inline ở dạng block.

**Display: inline;** → cho phép hiển thị một Element block ở dạng inline.

183

## Display

Ví dụ 1 :

```
<style>
p {color: red;}
p.ex1 {display: none;}
p.ex2 {display: inline;}
p.ex3 {display: block;}
</style>

<h2>display: none;</h2>
<div>
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper
diam at erat pulvinar, at pulvinar felis blandit. <p class="ex1">HELLO
WORLD!</p> Vestibulum volutpat tellus diam, consequat gravida libero
rhoncus ut.
</div>

<h2>display: inline;</h2>
<div>
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper
diam at erat pulvinar, at pulvinar felis blandit. <p class="ex2">HELLO
WORLD!</p> Vestibulum volutpat tellus diam, consequat gravida libero
rhoncus ut.
</div>

<h2>display: block;</h2>
<div>
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper
diam at erat pulvinar, at pulvinar felis blandit. <p class="ex3">HELLO
WORLD!</p> Vestibulum volutpat tellus diam, consequat gravida libero
rhoncus ut.
</div>
```

display: none;

Notice ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper
diam at erat pulvinar, at pulvinar felis blandit. HELLO WORLD! Vestibulum volutpat tellus diam, consequit
gravida libero rhoncus ut.

display: inline;

Notice ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper
diam at erat pulvinar, at pulvinar felis blandit. HELLO WORLD! Vestibulum volutpat tellus diam, consequit
gravida libero rhoncus ut.

display: block;

Notice ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper
diam at erat pulvinar, at pulvinar felis blandit. HELLO WORLD!

Vestibulum volutpat tellus diam, consequat gravida libero rhoncus ut.

184

## CSS-Shadow

Text-shadow: → cho phép thêm bóng đổ vào text

CP:

```
CSS selector {
 Text-shadow: h v b color,....;
}
```

```
<!DOCTYPE html>
<html>
<head>
<style>
h1 {
 text-shadow: 0 0 3px #FF0000, 0 0 5px #0000FF;
}</style>
</head>
<body>

<h1>Lớp 11DHBM1 nói chuyện nhiều!</h1>

</body>
</html>
```

Lớp 11DHBM1 nói chuyện nhiều!

185

## CSS-Shadow

Box-shadow → cho phép thêm bóng đổ vào Element

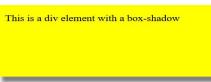
CP:

```
CSS selector {
 Box-shadow: h v b color,....;
}
```

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
 width: 300px;
 height: 100px;
 padding: 15px;
 background-color: yellow;
 box-shadow: 10px 10px 5px grey;
}</style>
</head>
<body>

<div>This is a div element with a box-shadow</div>

</body>
</html>
```



186

## CSS-Transition Transform

Transition → cho phép thay đổi thuộc tính trong một khoảng thời gian.

transition: width 2s, height 4s;

Transition-delay : 1s;

transition: width 2s, height 2s, transform 2s;

transform: rotate(180deg);



CSS

187

## CSS-Opacity

Opacity → Thường được sử dụng với pseado hover cho phép thay đổi độ trong suốt của đối tượng khi lướt chuột qua.



188

## CSS-Transform

### Các Method

Function	Description
matrix( <i>n,n,n,n,n,n</i> )	Defines a 2D transformation, using a matrix of six values
translate( <i>x,y</i> )	Defines a 2D translation, moving the element along the X- and the Y-axis
translateX( <i>n</i> )	Defines a 2D translation, moving the element along the X-axis
translateY( <i>n</i> )	Defines a 2D translation, moving the element along the Y-axis
scale( <i>x,y</i> )	Defines a 2D scale transformation, changing the elements width and height
scaleX( <i>n</i> )	Defines a 2D scale transformation, changing the element's width
scaleY( <i>n</i> )	Defines a 2D scale transformation, changing the element's height
rotate( <i>angle</i> )	Defines a 2D rotation, the angle is specified in the parameter
skew( <i>x-angle,y-angle</i> )	Defines a 2D skew transformation along the X- and the Y-axis
skewX( <i>angle</i> )	Defines a 2D skew transformation along the X-axis
skewY( <i>angle</i> )	Defines a 2D skew transformation along the Y-axis

189

## CSS-Animation

→ Cho phép diễn hoạt các Element trong HTML mà không cần sử dụng Javascript hay Flash.

### Các thuộc tính

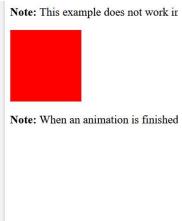
- `@keyframes`
- `animation-name`
- `animation-duration`
- `animation-delay`
- `animation-iteration-count`
- `animation-direction`
- `animation-timing-function`
- `animation-fill-mode`
- `animation`

190

### Ví dụ 1:

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
 width: 100px;
 height: 100px;
 background-color: red;
 animation-name: example;
 animation-duration: 4s;
}

@keyframes example {
 from {background-color: red;}
 to {background-color: yellow;}
}
</style>
</head>
```

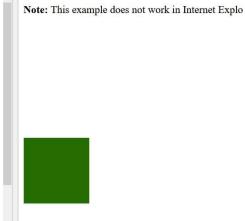


191

### Ví dụ 2:

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
 width: 100px;
 height: 100px;
 background-color: red;
 position: relative;
 animation-name: example;
 animation-duration: 5s;
 animation-timing-function: linear;
 animation-delay: 2s;
 animation-iteration-count: infinite;
 animation-direction: alternate;
}

@keyframes example {
 0% {background-color:red; left:0px; top:0px;}
 25% {background-color:yellow; left:200px; top:0px;}
 50% {background-color:blue; left:200px; top:200px;}
 75% {background-color:green; left:0px; top:200px;}
 100% {background-color:red; left:0px; top:0px;}
}
</style>
```



192

## Cách viết thu gọn các thuộc tính Animation

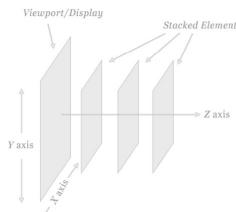
```
div {
 animation-name: example;
 animation-duration: 5s;
 animation-timing-function: linear;
 animation-delay: 2s;
 animation-iteration-count: infinite;
 animation-direction: alternate;
}

div {
 animation: example 5s linear 2s infinite alternate;
}
```

193

## Z-index

Index gồm 3 tọa độ là x, y, z. Với x, y là tọa độ mặt phẳng, có thể nói nó là width và height trong css. z là trục không gian để tạo ra cấu trúc 3D.



194

## Z-index

- Thuộc tính z-index thiết lập thứ tự xếp chồng nhau của một thành phần vị trí. Thứ tự chồng nhau được sắp xếp dựa theo giá trị số, thành phần HTML nào có chỉ số z-index cao hơn sẽ nằm trên, ngược lại sẽ nằm dưới, giá trị mặc định là 0, có thể sử dụng số âm. Giá trị tốt nhất là không sử dụng đơn vị.
- Chú ý: z-index chỉ làm việc cùng với thuộc tính position.

195

## Z-index

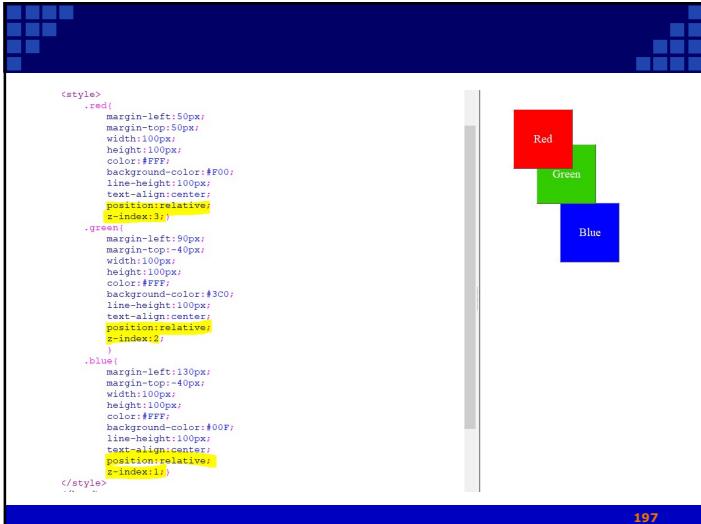
- Ví dụ: cho 3 block hình vuông như sau

```
<style>
.red{
margin-left:10px;
margin-top:10px;
width:100px;
height:100px;
color:#FF0000;
background-color:#FF0000;
line-height:100px;
text-align:center;
}.green{
margin-left:10px;
margin-top:10px;
width:100px;
height:100px;
color:#008000;
background-color:#008000;
line-height:100px;
text-align:center;
}.blue{
margin-left:10px;
margin-top:10px;
width:100px;
height:100px;
color:#0000FF;
background-color:#0000FF;
line-height:100px;
text-align:center;
}</style>
</head>
<body>
<div class="red">Red</div>
<div class="green">Green</div>
<div class="blue">Blue</div>
</body>
```



Hãy sử dụng thuộc tính z-index để thay đổi thứ tự xếp chồng của các block hình vuông như sau

196



## Grid Layout

Grid Layout offers a grid-based layout system, with rows and columns, making it easier to design web pages without having to use float and positioning

198



TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THỰC PHẨM THÀNH PHỐ HỒ CHÍ MINH  
KHOA CÔNG NGHỆ THÔNG TIN

# THIẾT KẾ WEB

BỘ MÔN KHOA HỌC MÁY TÍNH

200

## Chương 4.

# JAVASCRIPT

1. Giới thiệu JavaScript
2. Cách sử dụng JavaScript
3. Các thành phần của Javascript
4. Một số đối tượng trong Javascript
5. Xử lý sự kiện trong Javascript

201

## 1. Giới thiệu Javascript

- JavaScript là ngôn ngữ lập trình phổ biến nhất thế giới
- JavaScript là ngôn ngữ lập trình cho Web
- JavaScript dễ học
- JavaScript có thể thay đổi HTML Content

```
<body>
<h2>JavaScript có thể thay đổi HTML Content</h2>
<p id="demo">JavaScript có thể thay đổi HTML Content</p>
<button type="button" onclick="document.getElementById('demo').innerHTML='Hello
JavaScript!'> Click Me to Change Content!</button>
</body>
```

### JavaScript có thể thay đổi HTML Content

Hello JavaScript!

[Click Me to Change Content!](#)

202

## 1. Giới thiệu Javascript

- JavaScript có thể thay đổi giá trị thuộc tính thẻ HTML

```
<body>
<h2>JavaScript có thể thay đổi giá trị thuộc tính trong HTML</h2>
<button type="button" value="TURN ON THE LIGHT" style="height:50px; background-color:#00F; color:white;" onclick=
"document.getElementById('myImage').src='Images/lighton.JPG'> />

<input type="button" value="TURN OFF THE LIGHT" style="height:50px; background-color:#F00; color:FFF;" onclick=
"document.getElementById('myImage').src='Images/lightoff.JPG'> />
</body>
```



203

## 2. Cách sử dụng

- Trong HTML, javascript được đặt trong cặp thẻ `<script>` và `</script>`.
- Các hàm trong Javascript là một khối mã lệnh mà có thể được thực thi khi nó được gọi. (một hàm có thể được gọi khi một sự kiện xảy ra ví dụ người dùng click vào button )
- Hàm JavaScript có thể được đặt trong `<body>` hoặc trong `<head>`

```
<!DOCTYPE html>
<html>
<head>
<script>
function myFunction() {
 document.getElementById("demo").innerHTML = "Paragraph changed.";
}
</script>
</head>
<body>
<h1>A Web Page</h1>
<p id="demo">A Paragraph</p>
<button type="button" onclick="myFunction()">Try it</button>

<script>
function myFunction() {
 document.getElementById("demo").innerHTML = "Paragraph changed.";
}
</script>
</body>
</html>
```

204

## 2. Cách sử dụng

- Javascript có thể được tổ chức trong một file độc lập bên ngoài `myScript.js`

```
function myFunction() {
 document.getElementById("demo").innerHTML = "Paragraph changed.";
}
```
- Sử dụng đoạn script bên ngoài

```
<script src="myScript.js"></script>
```
- Một số ích lợi của external Script
  - Tách biệt code HTML và JS
  - Dễ đọc hơn và dễ quản lý hơn
  - Các file js có thể được cache giúp page load nhanh hơn.

205

## 3. Các thành phần của JavaScript

- JavaScript Output
- Syntax
- Conditional Expression
- JS Loop (For; For in; While; break)

206

### 3.1. JavaScript Output

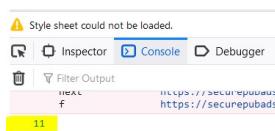
- Sử dụng `document.write()`

```
<script>
document.write(5 + 6);
</script>
```
- Sử dụng `window.alert()`

```
<script>
window.alert(5 + 6);
</script>
```


- Sử dụng `console.log()`

```
<script>
console.log(5 + 6);
</script>
```



207

### 3.2. Syntax

- Kết thúc câu lệnh JavaScript có thể có dấu chấm phẩy `(;)` hoặc không. Tuy nhiên, khuyến cáo luôn sử dụng.
- Tạo chú thích

```
<script>
// Comment một dòng lệnh
/*
Comment một khối lệnh
*/
function myFunction() {
 document.getElementById("panel").style.display = "block";
}
```

208

## 3.2. Syntax

- Các kiểu dữ liệu cơ bản

Kiểu dữ liệu	Ví dụ	Giải thích
Kiểu số	3 8.7 -17	Biểu diễn kiểu số nguyên và số thực
Kiểu chuỗi	"Hello World" '100'	Biểu diễn chuỗi, có thể sử dụng dấu nháy đơn hoặc kép để biểu diễn chuỗi trong js
Boolean	true false	Biểu diễn dữ liệu đúng hoặc sai
Null	null	đại diện cho giá trị không có gì trong js
undefined	undefined	tương tự, nhưng biểu thị cho giá trị chưa được định nghĩa

209

## 3.2. Syntax

- Biến và hằng

Khai báo	Ví dụ	Giải thích
let	let test; let test = 5;	Khai báo biến test, và nó mang giá trị undefined. Khai báo biến test và gán cho nó giá trị 5. Biến khai báo bằng let có phạm vi trong một block {}
var	var test; var test = 5;	Tương tự như let. Tuy nhiên, biến khai báo bằng var có phạm vi trong một hàm.
const	const test = 5;	Khai báo hằng số

210

## 3.3. Câu lệnh rẽ nhánh

- So sánh. Giả sử:  $x = 5$

Toán tử	Tên	Ví dụ	Kết quả
>	lớn hơn	$x > 20$	false
$\geq$	lớn hơn hoặc bằng	$x \geq 5$	true
<	nhỏ hơn	$x < -100$	false
$\leq$	nhỏ hơn hoặc bằng	$x \leq 25$	true
$\equiv$	bằng (tự động thay đổi kiểu dữ liệu)	$x \equiv '5'$	true
$\neq$	không bằng	$x \neq 'a'$	true
$\equiv\equiv$	bằng (kiểm tra cả giá trị và kiểu dữ liệu)	$x \equiv\equiv '5'$	false
$\neq\neq$	khác giá trị hoặc kiểu dữ liệu	$x \neq\neq '5'$	true

211

## 3.3. Câu lệnh rẽ nhánh

- Toán tử Logic. Giả sử:  $x = 5$  và  $y = 10$

Toán tử	Tên	Ví dụ	Kết quả
$\&\&$	AND	$x < 20 \&\& x \neq 5$	false
$\  \ $	OR	$y > 10 \  x \equiv\equiv 5$	true
!	NOT	$! (x \equiv\equiv y)$	true

212

### 3.3. Câu lệnh rẽ nhánh

- Câu lệnh if

```
if (conditional1){
 // Khối lệnh 1 được thực thi nếu conditional là true
} else if (conditional2){
 // Khối lệnh 2;
} else {
 // Khối lệnh 3
}
```

- Ví dụ

```
let age = 18;
if (age < 18) {
 alert("Bạn chưa đủ tuổi");
} else {
 alert("Mời vào !!!");
}
```

213

### 3.3. Câu lệnh rẽ nhánh

- Switch được sử dụng để biểu diễn các hành động khác nhau dựa trên các điều kiện khác nhau

```
switch (expression)
{
 case x: // code block;
 break;
 case y: // code block;
 break;
 default:
 // code block
}
```

214

### 3.4. Vòng lặp

- Vòng lặp for

```
for ([initialization]; [condition]; [final-expression])
 statement
```

- Ví dụ

```
let str = '';

for (let i = 0; i < 9; i++) {
 str = str + i;
}

console.log(str);
// output: "012345678"
```

215

### 3.4. Vòng lặp

- Vòng lặp for ... of

```
for (variable of iterable) {
 statement
}
```

- Ví dụ

```
const iterable = [10, 20, 30];

for (let value of iterable) {
 value += 1;
 console.log(value);
}
// 11
// 21
// 31
```

216

## 3.4. Vòng lặp

- Vòng lặp for ... in

```
for (variable in object) {
 statement
}
```

- Ví dụ

```
var obj = {a: 1, b: 2, c: 3};

for (const prop in obj) {
 console.log(prop);
}

// Output:
// a
// b
// c
```

217

## 3.4. Vòng lặp

- Vòng lặp while

```
while (condition)
 statement
```

- Ví dụ

```
var n = 0;
var x = 0;

while (n < 3) {
 n++;
 x += n;
}
```

218

## 3.4. Vòng lặp

- Vòng lặp do ... while

```
do
 statement
while (condition);
```

- Ví dụ

```
var result = '';
var i = 0;
do {
 i += 1;
 result += i + ' ';
}
while (i > 0 && i < 5);
// Mặc dù i == 0 vòng lặp vẫn thực hiện trước khi thực hiện kiểm tra
```

219

## 3.4. Vòng lặp

- break: thoát khỏi vòng lặp ngay lập tức

- Ví dụ

```
let i = 0;

while (i < 6) {
 if (i === 3) {
 break;
 }
 i = i + 1;
}

console.log(i);
// output: 3
```

220

## 3.4. Vòng lặp

- continue: dừng vòng lặp hiện tại và thực hiện vòng lặp mới.
- Ví dụ

```
let text = '';
for (let i = 0; i < 10; i++) {
 if (i === 3) {
 continue;
 }
 text = text + i;
}

console.log(text);
// output: "012456789"
```

221

## 3.5. Hàm

- Khai báo hàm

```
function fnName(param1 = defaultValue1, ..., paramN = defaultValueN) { /* ... */ }
```

- Ví dụ

```
function multiply(a, b = 1) {
 return a * b
}

multiply(5, 2) // 10
multiply(5) // 5
```

222

## 3.6. Mảng

- Tạo mảng

```
let fruits = ['Apple', 'Banana']

console.log(fruits.length)
// 2
```

- Truy suất mảng sử dụng chỉ số

```
let first = fruits[0]
// Apple

let last = fruits[fruits.length -
1]
// Banana
```

223

## 3.6. Mảng

- Lặp qua các phần tử

```
fruits.forEach(function(item,
index, array) {
 console.log(item, index)
})
// Apple 0
// Banana 1
```

224

## 3.6. Mảng

- Một số phương thức

Phương thức	Điễn giải	Ví dụ
push	Thêm phần tử vào cuối mảng	fruits.push('Orange')
pop	Xóa phần tử cuối mảng	fruits.pop()
shift	Xóa phần tử đầu mảng	fruits.shift()
unshift	Thêm phần tử đầu mảng	fruits.unshift('Strawberry')
indexOf	Tìm chỉ số phần tử của mảng	fruits.indexOf('Banana')
splice	Xóa phần tử bằng chỉ số	fruits.splice(index, 1)
slice	Copy một mảng	fruits.slice()

225

## 4. Thao tác với DOM

- DOM (Document Object Model) là giao diện cho phép sử dụng JavaScript với HTML và CSS. Trình duyệt biến mọi phần tử HTML thành một đối tượng JavaScript được lưu trữ trong đối tượng tài liệu và có thể truy cập trong mã JavaScript.
- Ví dụ: chọn phần tử thẻ `<a>` đầu tiên và thay đổi đường link trong tài liệu HTML

```
let link = document.querySelector('a');
link.href = 'http://thongthai.work';
```

226

## 4. Thao tác với DOM

- Một số phương thức chọn phần tử DOM

Phương thức	Điễn giải
<code>var element = document.getElementById(id);</code>	Chọn phần tử theo ID
<code>var elements = document.getElementsByTagName(name);</code>	Chọn <b>tất cả</b> các phần tử theo Tên thẻ HTML
<code>var elements = document.getElementsByClassName(names);</code>	Chọn <b>tất cả</b> các phần tử theo Tên class
<code>var element = document.querySelector(selectors);</code>	Chọn phần tử đầu tiên thỏa theo các thức chọn CSS Selector
<code>var elementList = parentNode.querySelectorAll(selectors);</code>	Chọn <b>tất cả</b> phần tử thỏa theo các thức chọn CSS Selector

227

## 4. Thao tác với DOM

- Tạo/Xóa Node HTML

```
var newDiv = document.createElement('div');
newDiv.innerHTML = 'Hello there!';
document.body.appendChild(newDiv);
```

```
document.body.removeChild(div1);
```

228

## 4. Thao tác với DOM

- Quản lý style với DOM

```
para.style.color = 'white';
para.style.backgroundColor = 'black';
para.style.padding = '10px';
para.style.width = '250px';
para.style.textAlign = 'center';
```

229

## 4. Thao tác với DOM

- Xử lý Sự kiện

```
var button1 = document.getElementById('button1');
button1.addEventListener('click', function(event){
 target = event.target;
 alert(target.innerHTML);
});
```

- Xem thêm các sự kiện tại:  
[https://www.w3schools.com/jsref/dom\\_obj\\_event.asp](https://www.w3schools.com/jsref/dom_obj_event.asp)

230

## 4. Thao tác với DOM

- Page Load Event

```
document.addEventListener('DOMContentLoaded', function(event){
 alert('Page has been loaded');
});
```

- MouseOver Event

```
document.getElementById('div1').addEventListener('mouseover',
function(event) {
 event.target.style.backgroundColor = '0000ff';
});
```

231

## 4. Thao tác với DOM

- Điều khiển form

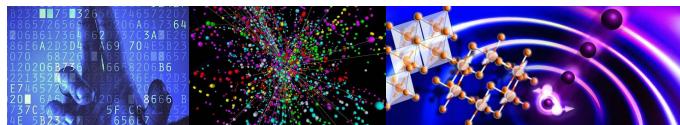
```
<form id='form'>
 Name: <input type='text'>
 <input type='submit' value='Save'>
</form>
```

```
var form = document.getElementById('form');
form.addEventListener('submit', function(event){
 event.preventDefault();
 var textInput = form.elements[0]
 alert(textInput.value)
});
```

232

## THIẾT KẾ WEB

BỘ MÔN KHOA HỌC MÁY TÍNH



### Chương 5.

## BOOTSTRAP 5

1. Giới thiệu Bootstrap
2. Cách cài đặt Bootstrap
3. Bootstrap Grid System
4. Bootstrap Component

234

### 1. Giới thiệu Bootstrap

- Bootstrap là gì ?
  - Framework mã nguồn mở cho việc phát triển website và webapp
  - Dựa trên HTML và CSS
  - Cung cấp một số template định dạng sẵn cho các layout và các phần tử (hệ thống grid system và các component)

235

### 1. Giới thiệu Bootstrap

- Tại sao sử dụng Bootstrap ?
  - Giúp việc thiết kế web dễ dàng và nhanh chóng hơn
  - Có thể nhanh chóng xây dựng một trang web “dễ nhìn”
  - Các phần tử được định dạng sẵn để “dễ nhìn” rồi !!!
  - VD: Một button có thể có sẵn một màu, bo tròn góc, font chữ đẹp, hiệu ứng rê chuột, ...

236

## 1. Giới thiệu Bootstrap

- Sử dụng Bootstrap như thế nào ?
  1. Thêm những đoạn mã cần thiết vào phần header của trang HTML
  2. Tìm các phần tử mong muốn trên website bootstrap và copy-paste đoạn code.
  3. Tùy biến phần tử
  4. Tùy biến thêm với CSS

237

## 2. Cách cài đặt Bootstrap

- Bootstrap có thể được sử dụng ngay mà không cần cài đặt nhờ sự hỗ trợ của các CDN (**content delivery network**). Xem địa chỉ các CDN cần thiết tại: <https://getbootstrap.com/docs/5.1/getting-started/introduction/>

```
<html>
 <head>
 <meta charset="utf-8">
 <meta name="viewport" content="width=device-width, initial-scale=1">

 <!-- Import Bootstrap CSS -->
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-1BmE4kwBq78iyHf1dVKuhfTAU6au8tT94WHftjDbrCEXSU1oBody12QvZ6jIW3" crossorigin="anonymous">
 </head>

 <body>
 <!-- Import Bootstrap JS -->
 <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.10.2/dist/umd/popper.min.js" integrity="sha384-7+zCNj/IqJ95wo160MtsKbz9ccEh31e0z1H6yDuCQ6gvnyJNSYdrPa03rtR1zdB" crossorigin="anonymous"></script>
 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.min.js" integrity="sha384-QJHtvGhmr9XKOipI6VutG+2QK9T4znN4kzFN1Rtk3zEFE1sxhlw15/YESvpZ13" crossorigin="anonymous"></script>
 </body>
</html>
```

238

## 3. Bootstrap Grid system

- Breakpoint

Breakpoint	Class infix	Dimensions
X-Small	None	<576px
Small	sm	≥576px
Medium	md	≥768px
Large	lg	≥992px
Extra large	xl	≥1200px
Extra extra large	xxl	≥1400px

239

## 3. Bootstrap Grid system

- Containers

	Extra small <576px	Small ≥576px	Medium ≥768px	Large ≥992px	X-Large ≥1200px	XX-Large ≥1400px
.container	100%	540px	720px	960px	1140px	1320px
.container-sm	100%	540px	720px	960px	1140px	1320px
.container-md	100%	100%	720px	960px	1140px	1320px
.container-lg	100%	100%	100%	960px	1140px	1320px
.container-xl	100%	100%	100%	100%	1140px	1320px
.container-xxl	100%	100%	100%	100%	100%	1320px
.container-fluid	100%	100%	100%	100%	100%	100%

240

### 3. Bootstrap Grid system

- Cơ bản Grid system

- ❖ Grid System gồm các hàng và cột (row & col)

```
<!-- Correct -->
<div class="row">
 <div class="col-md-12">Content Goes Here</div>
</div>
```

```
<!-- Wrong -->
<div class="row">
 Some content
</div>
```

- ❖ Hàng (row) được chia thành 12 phần bằng nhau

Hàng được chia thành 12 phần bằng nhau. Khi chúng ta đặt một cột trong một hàng, chúng ta phải xác định số lượng của các bộ phận nó sẽ chiếm.

col-md-NUMBER. NUMBER là một số nguyên từ 1 đến 12. Tùy thuộc vào số lượng, một cột sẽ chiếm một tỷ lệ phần trăm của chiều rộng của hàng: 6 cột sẽ là 50% (6/12), 3 sẽ là 25% (3/12)

```
<div class="row">
 <div class="col-md-12">Full width 100%</div>
</div>
<div class="row">
 <div class="col-md-3">25%</div>
 <div class="col-md-3">25%</div>
 <div class="col-md-6">50%</div>
</div>
```

241

### 3. Bootstrap Grid system

- Cơ bản Grid system

- ❖ Cột (col)

Chỉ có 12 chỗ trống có sẵn trong một hàng. Nếu chúng ta tổng hợp các không gian theo yêu cầu của một vài cột liền kề, và kết quả vượt quá 12, cột cuối cùng trong nhóm đó sẽ phải di chuyển đến dòng kế tiếp.

```
<div class="row">
 <div class="col-xs-8"></div>
 <div class="col-xs-4"></div>
 <div class="col-xs-9">This column will move to the next line.</div>
</div>
```

242

### 4. Bootstrap Component

- Xem các component tại <https://getbootstrap.com/docs/5.1/components/>

243